UNIVERSITÉ DE SHERBROOKE
Faculté de génie
Département de génie électrique et de génie informatique

Transport collaboratif d'une charge par deux robots humanoïdes

Mémoire de maitrise
Spécialité : génie électrique

Louis Hawley

Sherbrooke (Québec) Canada

Septembre 2018

MEMBRES DU JURY

Wael Suleiman

Directeur

Nicolas Quaegebeur

Évaluateur

Karim Bouyarmane

Évaluateur

RÉSUMÉ

La structure bipède des robots humanoïdes leur confère une grande agilité et la capacité de se déplacer dans des environnements encombrés qui ne sont pas accessibles à des robots à roues plus traditionnelles. Cette particularité fait en sorte que ce type de robot est le mieux adapté pour évoluer dans des environnements conçus pour et par l'homme. Cette grande agilité a toutefois un prix puisque les humanoïdes sont plus complexes à contrôler étant donné l'instabilité inhérente à la marche bipède. Dans ce projet de recherche, on s'intéresse au contrôle de robots humanoïdes dans le cadre de tâches très communes et intéressantes à reléguer aux robots, soit le transport d'objet. Le cas d'intérêt est le transport collaboratif d'une charge par deux humanoïdes étant donné que ça ne nécessite aucun outil externe et est ainsi applicable en toute circonstance.

En premier lieu, un estimateur d'état applicable pour les robots humanoïdes de petite taille est proposé, permettant ainsi d'estimer les interactions entre le robot et son environnement. Ensuite, une stratégie de contrôle permettant à un humanoïde d'utiliser un chariot de transport pour déplacer un objet lourd est présentée.

Finalement, le transport collaboratif par deux robots humanoïdes est abordé. Le système développé utilise un contrôleur externe qui planifie la trajectoire des robots et valide la stabilité des déplacements à l'aide d'un modèle dynamique simple du système basé sur des pendules inversés. Tous les algorithmes développés ont été validés et testés sur des robots humanoïdes NAO. Les résultats démontrent qu'il est possible de transporter un objet lourd sans modifier les composantes matérielles des robots, soit en utilisant un chariot ou bien en coopérant avec un autre robot. Les résultats obtenus pourraient s'avérer utiles dans certaines situations réelles telles que les tâches de manutention dans un domaine manufacturier ou bien le transport de blessé sur une civière.

Mots-clés : robots humanoïdes, tâches collaboratives, navigation, contrôle

TABLE DES MATIÈRES

LISTE DES FIGURES

CHAPITRE 1

INTRODUCTION

1.1 Contexte

Le 11 mars 2011, à Fukushima au Japon, a eu lieu le deuxième plus important accident nucléaire de l'histoire. Le Japon, qui était à l'époque et est encore aujourd'hui un chef de file dans le domaine de la robotique, n'a pas réussi à fournir aux ingénieurs des robots assez performants pour accéder aux zones contaminées des réacteurs et ainsi limiter les dommages. Un des problèmes était que les robots disponibles étaient surtout des bases mobiles à roues qui étaient peu appropriées pour l'environnement de la centrale. Il devenait clair que le besoin d'avoir des robots plus mobiles et capables de fonctionner dans des environnements complexes était bien réel.

L'année suivante, en réponse à Fukushima, le DARPA (Defense Advanced Research Projects Agency) lance le Darpa Robotics Challenge, un concours public visant la création de robots assez robustes et agiles pour évoluer dans des environnements trop dangereux pour l'homme comme des lieux de catastrophes naturelles. Il s'avère que les robots de forme humanoïde sont plus appropriés que les robots typiques à roues dans ce genre de situation en raison de leur anatomie. L'affiche officielle du DARPA challenge présenté ci-dessous montre certaines tâches qui pourraient être demandés à ces robots telles que l'utilisation d'outils ou l'actionnement de valve.

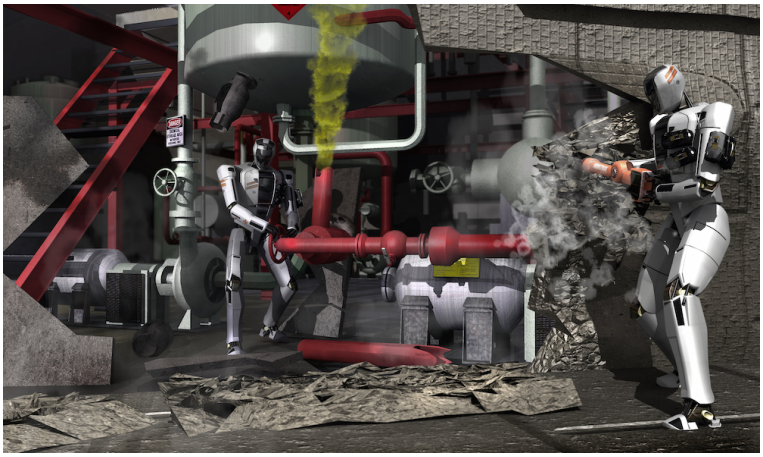


Figure 1.1 Affiche du DARPA Robotics Challenge

1.2 Robotique humanoïde

L'utilisation dans des situations d'urgence n'est toutefois pas le seul domaine où l'on s'intéresse à la robotique humanoïde. Entre autres, Honda vise la création d'un robot capable d'aider des personnes handicapées ou malades avec le développement du célèbre robot ASIMO[Sakagami *et coll.*, 2002]. Également, les humanoïdes sont utilisés depuis 2006 dans Robocup [Kitano *et coll.*, 1997], une compétition de soccer robotiques annuelles. Cette compétition stimule le développement de robot humanoïde de toute taille en offrant l'opportunité à des équipes provenant de partout dans le monde de tester leurs algorithmes d'intelligence artificielle et de contrôle dans le cadre d'une partie de soccer.

Que ce soit pour une utilisation dans un cadre domestique ou dans des situations d'urgences, on pourrait souhaiter voir ces robots effectuer des tâches de manipulation et de transport d'objets. Par exemple, pour l'évacuation d'une victime sur une civière ou pour aider une personne âgée à déplacer un gros objet. Toutefois, la charge que ces robots peuvent soulever est souvent faible par rapport aux robots à roue et est limitée par la puissance des actionneurs et par la structure bipède. Même s'il est normalement possible d'utiliser des actionneurs plus puissants, ce n'est pas toujours intéressant en raison du poids et des coûts supplémentaires qui en résultent. Une solution envisageable serait de faire coopérer plusieurs humanoïdes pour effectuer le transport d'un objet lourd.

1.3 Cadre du projet

Ce projet de recherche est la continuation de [Rioux, 2016] qui définissait un algorithme de planification de mouvement et de synchronisation des déplacements pour un humanoïde déplaçant une charge sur un chariot ainsi que pour une équipe de deux robots humanoïdes transportant une table. Cette fois-ci, la question n'est pas de savoir comment planifier le déplacement de deux humanoïdes transportant une charge mais plutôt comment s'assurer qu'ils effectuent la tâche de façon stable. Ainsi, l'emphase est portée sur l'étude de la dynamique alors que chacune des tâches est modélisée et des contrôleurs sont conçus permettant à un robot de déplacer de façon stable l'équivalent de son propre poids à l'aide d'un chariot et à une équipe de deux humanoïdes de transporter une charge de 2.13 Kg ($\sim$ 50% du poids d'un robot) en utilisant une table miniature. D'ailleurs, même si les résultats sont généralement applicables à tous les humanoïdes, une grande importance a été portée à l'implémentation sur des robots réels. La plate-forme utilisée dans le cadre du projet de recherche est le robot Nao, présenté à la Fig. 1.2. Le robot Nao est un humanoïde d'une hauteur d'environ 573 mm pour un poids d'environ 5.1 kg

fabriqué par Softbank Robotics. Il possède 25 degrés de liberté dont 5 dans chaque jambe et bras en plus d'être munis d'une centrale inertielle dans le torse et d'encodeur de position sur chacun des moteurs.

Figure 1.2 Un robot Nao utilisé dans le cadre du projet

Le reste du document est séparé en 5 sections distinctes. Au chapitre 2, une synthèse des recherches réalisés en liens avec la problématique est présentée. Les chapitres 3, 4 et 5 présentent les travaux qui ont été réalisés dans le cadre de ce projet. Plus précisément, le chapitre 3 aborde le défi d'estimer l'état d'un petit robot humanoïde qui, contrairement aux humanoïdes plus dispendieux de plus grande taille, n'est pas équipé de capteurs et d'actuateurs performants. Au chapitre 4, on s'intéresse au contrôle d'un robot humanoïde qui effectue une tâche de transport à l'aide d'un chariot mobile. Le chapitre 5 présente une stratégie de contrôle pour le transport coopératif d'un objet lourd par deux robots humanoïdes. Finalement, le chapitre 6 résume le projet et relate les contributions originales qui en découlent.

CHAPITRE 2

État de l'art

2.1 Contrôle de robots humanoïdes

2.1.1 Cadre de référence

La génération de marche d'un robot humanoïde peut être réalisé avec diverses approches tel que l'utilisation d'un réseau locomoteur spinal(*CPG*)[Kim et Lee, 2007] ou avec des algorithmes génétiques[Kambayashi *et coll.*, 2005]. Toutefois, l'approche la plus populaire est une conception de marche basée sur un modèle dynamique(*model-based design*). Dans ce cas, le robot est habituellement approximé par une version linéarisée d'un pendule inversé où le déplacement du pendule est limité dans le plan x-y (Fig. 2.1). Ainsi, l'équation de la dynamique de mouvement de ce système est :

$$M_c \, \ddot{x}_c = \frac{M_c \, g \, x_c}{Z_c} \tag{2.1}$$

où M_c est la masse du robot, Z_c est la hauteur du centre de masse, g est l'accélération gravitationnelle, x_c et $\ddot{x}_c$ sont respectivement la position et l'accélération de la projection du centre de masse sur le sol.

À partir de ce système simple, il est possible de générer une marche stable en contrôlant la position du ZMP (*Zero-moment point*). Le ZMP est défini comme le point au sol où le moment inertiel et le moment induit par la gravité s'annulent[Vukobratović et Borovac, 2004]. Plus simplement, ce point est équivalent au centre de pression. Il est couramment utilisé en robotique humanoïde comme critère de stabilité. En résumé, si le ZMP est situé dans le polygone de support du robot, la configuration est stable. Ainsi, d'après cette définition, la position du ZMP du système définit à l'équation 2.1 est :

$$x_{ZMP} = x_c - \frac{Z_c \, \ddot{x}_c}{g} \tag{2.2}$$

Un aspect problématique du projet est l'adaptation de la marche du robot en fonction de l'objet transporté et/ou des interactions avec son compagnon. En effet, le fait de transporter un objet lourd et/ou d'être lié de façon rigide à un autre robot risque de mo-

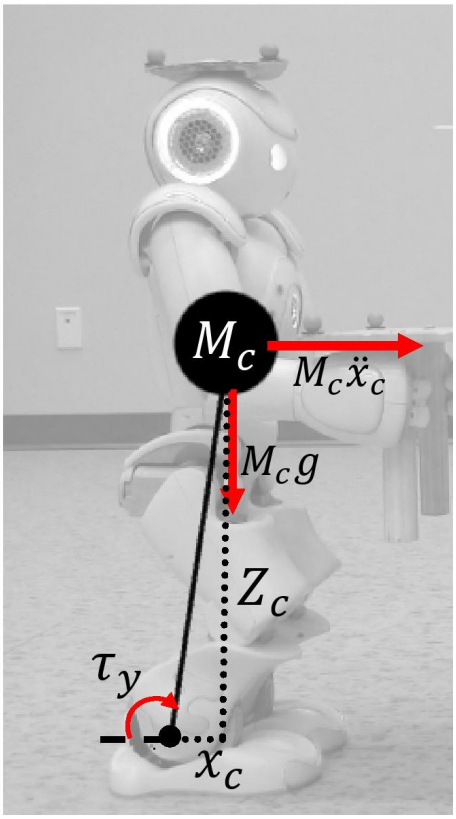


Figure 2.1 Modèle dynamique d'un humanoïde : Pendule inversé

difier la dynamique du système de façon importante et le modèle usuel du pendule inversé présenté plus tôt ne sera probablement plus assez représentatif pour assurer la stabilité des mouvements. Les sections suivantes présentent quelques travaux notables effectués dans le domaine de la génération de mouvement pour humanoïde lors d'interactions physiques avec des objets.

2.1.2 Tâches de transport par robot humanoide

Certaines recherches ont déjà été effectuées dans le but d'obtenir des modèles dynamiques alternatifs pour humanoïdes dans des situations d'interactions avec l'environnement. Entre autres, un robot HRP2 a été utilisé pour pousser un objet lourd au sol tout en marchant [Harada *et coll.*, 2003a]. Les auteurs ont révélé que si le modèle dynamique du pendule inversé n'était pas augmenté en considérant les interactions entre le robot et l'objet, le robot chutait rapidement. Dans ce cas-ci, le robot poussait l'objet avec les mains et l'interaction était mesurée directement à l'aide de capteurs de force/couple situé dans les poignets du robot. Une fois ces forces mesurées, il était possible de les introduire dans le modèle dynamique de la tâche et de générer une marche stable en analysant le ZMP.

Plus tard [Harada *et coll.*, 2005], la même approche à été utilisé pour qu'un humanoïde soulève et transporte une charge de façon stable.

Dans des travaux similaires [Nozawa *et coll.*, 2011a, 2008], la tâche analysée est plutôt le déplacement d'un sujet en fauteuil roulant par un robot HRP2. Au niveau du contrôle, l'approche est très similaire alors que le robot mesure les interactions avec la chaise à l'aide de capteurs de force et compense cette force d'interaction en modifiant le ZMP désiré. Les résultats obtenus montrent bien qu'il est possible de générer un déplacement stable lors du transport d'objet pourvu que les forces d'interactions puissent être mesurées. Or, le problème est que la performance est grandement dépendante des données recueillies par les capteurs de force aux poignets et que ce genre de capteurs est uniquement présent sur les grands robots dispendieux tels qu'Atlas et HRP-2. Ainsi, ces travaux démontrent la nécessité de trouver une façon de détecter les forces externes appliquées sur un robot humanoïde dépourvu de capteurs de force au point d'application de celle-ci. La sous-section suivante présente certaines pistes de solution pour détecter efficacement les forces externes sur des robots de plus petite taille telle que le Nao.

2.1.3 Stratégie de détection de force externe

Pour pouvoir assurer la stabilité du système lors de tâche de transport, il faut d'abord réussir à estimer les forces et perturbations exercées sur les robots. Pour cela, la centrale inertielle est une option intéressante étant donné qu'elle fait partie de l'ensemble de capteurs présent de façon standard sur les robots humanoïdes. Dans [Kaneko *et coll.*, 2012], un observateur de perturbation qui estime l'amplitude d'une force externe est présenté. L'observateur utilise des mesures provenant de la centrale inertielle et de capteurs de force/couple six-axes situés dans les chevilles pour estimer la grandeur d'une force externe importante. Par exemple, l'observateur estime des forces causées par un coup de pied appliqué sur le torse du robot et lors d'une collision avec l'environnement. Ainsi, la force externe était équivalente à une forte impulsion appliquée sur le système. Dans le cas d'un transport collaboratif, l'interaction entre les robots risque plutôt de créer une force constante avec certaines perturbations ponctuelles importantes. La performance de l'observateur risque donc de diminuer.

Certains travaux ont aussi été réalisés au niveau de l'estimation de force externe exercée sur un robot Nao. Dans [Berger *et coll.*, 2014, 2015], le Nao est utilisé pour effectuer une tâche de transport collaboratif avec un humain. La contribution principale est un algorithme d'apprentissage qui en utilisant les capteurs internes de Nao, peut détecter lorsqu'une force externe est appliquée sur le robot par son collègue humain. Par contre,

cette information n'est pas utilisée par le contrôleur du robot et sert seulement a guidé le déplacement de celui-ci. Dans [Mattioli et Venditteli, 2017], les auteurs utilisent plutôt le couple exercé par les moteurs pour estimer une force externe appliquée sur le robot. L'implémentation sur le robot Nao emploie plutôt l'erreur en position des moteurs pour estimer le couple étant donné que les moteurs de celui-ci ne possèdent pas de capteurs de couple. L'approche est seulement testée dans le cas statique et les résultats montrent que la friction statique sur les joints de Nao est trop élevée pour obtenir des résultats précis. Néanmoins, un grand avantage de la méthode est la possibilité d'estimer le point d'application de la force.

Une avenue possible pour estimer une force externe est l'utilisation de capteur de pression, un type de capteur couramment utilisé par les robots humanoïdes de petite taille. Les capteurs de pression (Force-sensing resistor (FSR)) du robot Nao sont présentés à la Fig. 2.2. Il consiste en une matrice de 4 capteurs de force qui permet de déduire la position du centre de pression sous chaque pied. Un aspect intéressant avec ce type de capteurs est qu'il permette d'estimer la position du ZMP tel que présenté dans [Shim *et coll.*, 2004] [Erbatur *et coll.*, 2002].

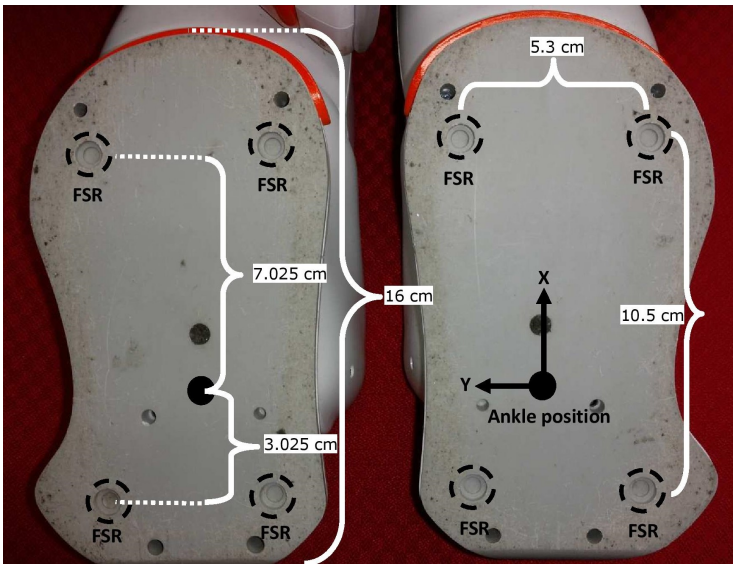


Figure 2.2 Capteurs de force sous les pieds du robot Nao

2.2 Transport collaboratif en robotique humanoïde

Dans [Yokoyama *et coll.*, 2003], le transport d'une table entre un robot HRP-2 et un humain est analysé. Les auteurs notent que l'utilisation des données recueillies par les capteurs de force dans les poignets du robot comme commande de mouvement au robot est problématique. En fait, lorsque l'humain tire ou pousse sur la table, le mouvement est

transmis au robot comme une perturbation qui le déstabilise. Pour éviter ce désagrément, les auteurs ont implémenté un contrôle des bras par impédance permettant ainsi de capter les forces appliquées sur les mains du robot sans les propager sur le centre de masse du robot. Bien que ce genre de contrôle ne soit pas directement applicable sur des robots ne disposant pas de capteur de force aux poignets, elle permet tout de même de valider que l'utilisation des bras comme amortisseur est envisageable.

Certains travaux ont aussi été réalisés sur des interactions entre deux humanoïdes[Wu *et coll.*, 2014]. Ici, une approche de type maitre-esclave est utilisé permettant le transport d'une objet rigide et léger. Pour avoir un déplacement stable, le robot esclave effectue des enjambées plus ou moins longues qui sont calculées avec une boucle de contrôle proportionnel-dérivée où la commande au système est fonction des forces mesurées par les capteurs de force aux poignets du robot. Également, un contrôle en impédance des bras est utilisé pour absorber les perturbations rapides. L'équation utilisée pour le contrôle en impédance est la suivante :

$$m\Delta\ddot{x} + d\Delta x + k\Delta x = F \tag{2.3}$$

où m est une inertie, d est un coefficient d'amortissement, k est un coefficient de ressort, Δx est le déplacement des mains par rapport à la valeur de référence et F est la force mesurée dans les poignets. À partir de cette équation, un contrôle en position des mains en fonction de la force mesurée est implémenté. Dans une recherche similaire [Wu *et coll.*, 2014, 2016], les mêmes auteurs ont modifiés l'architecture de contrôle en utilisant plutôt un contrôle centralisé au lieu de l'approche maitre-esclave. Le contrôleur utilisé est hybride et comprend une partie de contrôle en force et une autre partie en position. Dans la partie de contrôle en position, les variables de commandes sont la position du centre de masse de chacun des deux robots A et B ($p_{CoM_{AB}}$), l'orientation du torse de chacun des robots($R_{waist_{AB}}$), la position et l'orientation du pied en mouvement de chaque robot ($pR_{swingLeg_{AB}}$) et la position de l'objet transporté (z). La cinématique du système entier est alors donnée par :

$$\begin{bmatrix} \dot{p}_{CoM_{AB}} \\ \dot{R}_{waist_{AB}} \\ \dot{p}\dot{R}_{swingLeg_{AB}} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} diag(\mathbf{J}_{CoM_A}, \mathbf{J}_{CoM_B}) \\ diag(\mathbf{J}_{waist_A}, \mathbf{J}_{waist_B}) \\ diag(\mathbf{J}_{swingFoot_A}, \mathbf{J}_{swingFoot_B}) \\ \mathbf{J}_{quad} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1^A \\ \vdots \\ \dot{\theta}_{32}^A \\ \dot{\theta}_1^B \\ \vdots \\ \dot{\theta}_{32}^B \end{bmatrix} \tag{2.4}$$

où $\dot{\theta}_X^{AB}$ est l'ensemble des moteurs de chacun des deux robots HRP-2 utilisés dans cette recherche et **J** est la matrice jacobienne de la chaine cinématique particulière à chaque commande. Par exemple, la position du centre de masse est contrôlée par les moteurs de la jambe de soutien alors que la position du pied en mouvement est contrôlée par les moteurs de l'autre jambe. Le contrôle de la position de l'objet transporté utilise les moteurs des bras de chacun des deux robots. Les auteurs ont défini la matrice Jacobienne $\mathbf{J}_{quad}$ qui lie la vitesse des joints des bras des deux robots à la vitesse de l'objet transporté. La position du centre de masse et du pied en mouvement est directement obtenue en sortie du générateur de marche de chacun des robots alors que l'orientation du torse est tout simplement calculée pour qu'il reste le plus droit possible. La commande haut niveau est alors une trajectoire en position de l'objet transporté. Cet algorithme peut servir de point de départ pour ce projet. Toutefois, les résultats obtenus démontrent que ce contrôle en position accumule une erreur et produit des forces importantes sur chacun des robots lorsqu'ils se déplacent dans le plan sagittal. Pour compenser les erreurs et absorber les forces, un contrôle en force utilisant les capteurs de force/couple dans les poignets a été implémenté. Les détails du contrôle en force ne sont pas présentés ici étant donné qu'il nécessite que les robots soient équipés de capteurs de force aux poignets. Néanmoins, ça démontre la nécessité de développer un algorithme alternatif ne nécessitant pas ces capteurs dispendieux.

2.2.1 Coopération en robotique

Une des difficultés principales à contrôler un système multi-agents est la gestion du partage d'information entre tous les robots qui participe dans le but de développer une stratégie de contrôle. Ce problème a été abordé dans le cadre d'un projet visant à effectuer le transport d'objets imposants par des robots mobiles [Ota *et coll.*, 1995]. Les robots utilisés sont des robots à roues avec un bras manipulateur leur permettant de modifier la prise sur l'objet transporté. La stratégie globale utilisée consiste en une planification de trajectoires pour l'objet et une trajectoire individuelle pour chacun des robots. De cette façon, chacun des robots tente de manipuler l'objet pour maintenir la trajectoire et s'il détecte qu'il a accumulé une erreur trop importante, il relâche sa prise sur l'objet et laisse le contrôle aux autres robots le temps qu'il reprenne en place. Toutefois, l'implémentation nécessitait qu'un seul robot tienne fermement l'objet à la fois. Sinon, il y avait d'importants retours de force dans le cas où deux robots effectuaient des mouvements différents. Le système a toutefois été amélioré plus tard en centralisant le contrôle sur un seul robot[Miyata *et coll.*, 1997]. Ainsi, à tout moment, un robot se voit assigner le rôle de superviseur et il est responsable de planifier le déplacement. Les autres robots quant eux

maintiennent l'objet tout en respectant les consignes du robot superviseur. Un système de contrôle centralisé est très utile dans un scénario de collaboration étant donné que l'objectif est le même pour tous les robots. Ainsi, au lieu que chacun planifie le déplacement de la charge individuellement, une seule unité de calcul récupère l'ensemble des données disponibles et planifie la tâche pour tout le système. Un avantage secondaire est qu'il est possible d'avoir l'unité de calcul sur un serveur externe dans le cas où la puissance de calcul embarqué sur les robots est faible.

Une autre approche couramment utilisée pour le transport collaboratif d'objet est de type maitre-esclave sans communication. Un robot humanoïde de taille humaine, le HRP2[Kaneko *et coll.*, 2004], a été utilisé pour effectuer le transport d'une table en collaboration avec un humain en utilisant cette approche [Yokoyama *et coll.*, 2003]. Pour y arriver, l'algorithme implémenté utilisait les informations provenant de capteurs de force installés dans les poignets du robot. À partir des informations recueillies par les capteurs, le robot pouvait ainsi prédire le mouvement effectué par l'humain et choisir une action en conséquence. Les mouvements possibles étaient toutefois limités à des déplacements horizontal et vertical. Le plus grand problème avec cette approche est le délai entre le mouvement du maitre et la détection du mouvement par l'esclave. Même avec des capteurs aux poignets, soit très près du point d'application de la force, le déplacement devait s'effectuer à très basse vitesse. Pour une coopération entre robots qui ne possède pas de capteurs de force près des mains, la performance risque d'être dégradée de façon importante.

Dans la première itération de ce projet de maitrise[Rioux, 2016], il avait été possible de réaliser le transport d'un objet non rigide avec deux robots Nao. L'approche utilisée avait été de traiter les deux robots comme des unités indépendantes ayant chacun une trajectoire de référence à suivre. Fig. 2.3 montre le résultat du déplacement de deux robots transportant un objet en utilisant cette approche. Comme on peut le constater, l'imprécision de la marche bipède a comme effet d'engendrer des erreurs en position trop importante pour envisager d'utiliser le système avec un objet rigide. D'ailleurs, même en utilisant un module de flux optique pour détecter le mouvement relatif entre les robots, il n'était pas possible de maintenir la position relative des robots assez constante pour espérer avoir un transport coopératif stable.

2.2.2 Résumé

D'après l'analyse des travaux pertinents réalisés dans le domaine du transport coopératif en robotique, on peux diviser en 3 catégories les architectures de contrôle : Leader/Suiveur (Leader-follower), Maître/Esclave et MIMO(Multiple-Input Multiple-

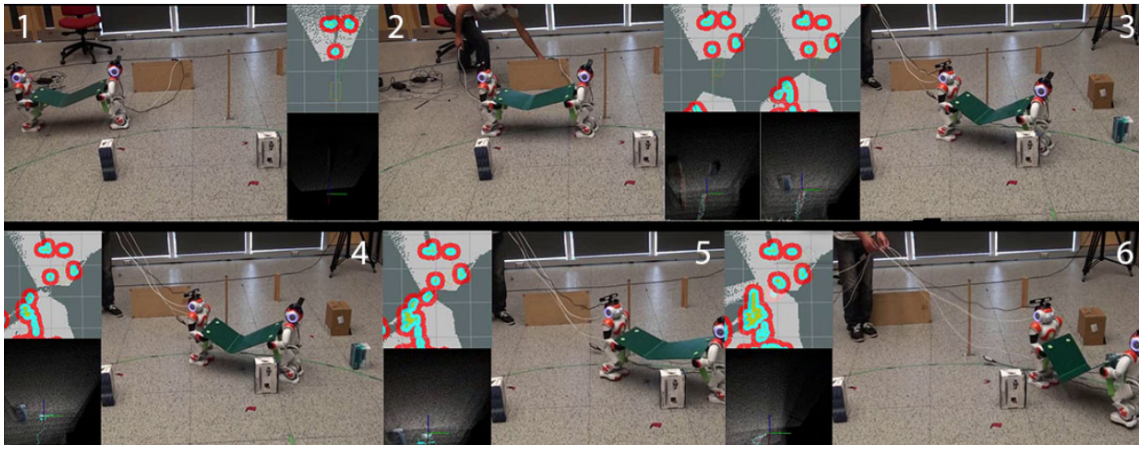


Figure 2.3 Robots effectuant une tâche de transport [Rioux *et coll.*, 2015]

Output). Les sous-sections suivantes résument chacune des architectures et discutent de leur applicabilité au projet.

Maitre/Esclave

Cette approche est la plus simple alors qu'elle consiste seulement à avoir un robot qui planifie le mouvement et se déplace tandis que les autres doivent se conformer au mouvement de celui-ci (voir Fig. 2.4). Le désavantage de cette architecture est qu'elle nécessite d'avoir des capteurs performants pour que le mouvement du maitre soit détecté rapidement par les esclaves. Également, elle ne permet pas d'effectuer des mouvements complexes étant donné qu'il n'y a pas de communication entre les agents. Étant donné l'instabilité des robots humanoïdes, l'approche est plus appropriée pour des interactions homme-robot alors que les humains peuvent rapidement s'ajuster au robot pour absorber les perturbations.

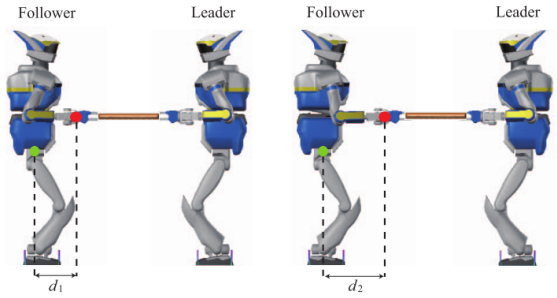


Figure 2.4 Coopération de type maitre-esclave [Wu *et coll.*, 2014]

Leader/Suiveur (*Leader/Follower*)

L'architecture Leader/Suiveur est l'objet de beaucoup de travail de recherche dans le domaine du vol en formation de drones ou engins spatiaux [Scharf *et coll.*, 2004].

Dans cette architecture, chacun des agents du système se voit assigner une trajectoire de référence qu'il doit suivre le mieux possible. Ainsi, l'ensemble du contrôle est réduit à une série de problèmes de suivi de trajectoires indépendant et distribué sur tous les robots. Habituellement, l'un des agents est dépendant de l'autre par l'une des façons suivantes : (1) l'agent j suit une trajectoire définit par une fonction reliant son état à l'état de l'agent i, (2) la trajectoire de référence de l'agent j est une fonction de l'état de i , ou (3) la commande d'asservissement de l'agent i est utilisé dans le contrôleur de j. L'avantage de cette approche par rapport à l'architecture maitre-esclave est que des trajectoires plus complexes peuvent être effectuées étant donné que les deux robots sont contrôlés indépendamment. Ainsi, ils peuvent se déplacer dans des endroits plus restreints avec des virages serrés. C'est cette approche qui a été utilisée dans la version antérieure de ce projet tel qu'expliqué dans la section précédente. Puisque la performance n'était pas suffisante en utilisant cette architecture, il est probable que le transport collaboratif entre humanoïdes nécessite un contrôle de précision qui ne peut être atteint en traitant les deux robots indépendamment. D'ailleurs, un des problèmes potentiels est que si la stabilisation de chaque robot est traitée indépendamment, la compensation effectuée par l'un des robots risque de se propager à l'autre via l'objet transporté et de le déstabiliser. Le second robot risque ensuite d'effectuer lui-même un mouvement pour compenser cette force qui sera de nouveau propagée vers son compagnon.

Entrées multiples, Sorties multiples

L'approche MIMO (*Multiple-Inputs Multiple-Outputs*) est elle aussi utilisée dans le domaine du vol en formation d'engins spatiaux[Scharf *et coll.*, 2004]. En utilisant cette architecture, on considère chaque véhicule/robot comme un élément d'un système à contrôler. Ainsi, la formation de robots/véhicules est traitée comme un système avec de multiples entrées et de multiples sorties. Pour cette raison, toutes les informations doivent être envoyées à une seule unité de calculs, qui peut être externe ou bien sur un des agents, qui planifie la tâche pour tout le système. L'avantage de cette architecture est qu'elle permet un contrôle optimal et stable. Par contre, elle est plus complexe à implémenter puisqu'elle nécessite d'avoir un modèle dynamique du système complet incluant les interactions entre les agents. Dans le cadre de ce projet, cela implique de développer un nouveau modèle dynamique pour la tâche de transport avec deux humanoïdes.

CHAPITRE 3

Estimateur de force externe pour des robots humanoïdes de petite taille

Auteurs et affiliation :

L.Hawley : Étudiant à la maîtrise en génie électrique, Université de Sherbrooke, Faculté de génie, Département de génie électrique et informatique

W.Suleiman : Professeur, Laboratoire de robotique intelligente, interactive, intégrée et interdisciplinaire (IntRoLab), Université de Sherbrooke, Faculté de génie, Département de génie électrique et de génie informatique

État de la publication : version finale publiée

Date de publication : Novembre 2016

Conférence : 2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)

Référence : Hawley, L., & Suleiman, W. (2016). External Force Observer for Medium-sized Humanoid Robots. Dans IEEE-RAS 16th International Conference on Humanoid Robots (pp. 366–371).

Titre français : Observateur de force externe pour robot humanoïde de taille moyenne

Contribution au document : La coopération entre deux robots humanoïdes est une tâche hautement dynamique et il est important de pouvoir mesurer ou estimer les interactions entre les robots si on veut être en mesure d'estimer l'état du système pour ensuite prendre les actions requises pour stabiliser le tout. La plate-forme disponible pour les tests, le robot Nao, n'est pas conçue pour effectuer ce genre de tâche étant donné que ce robot ne possède pas l'éventail de capteurs dont les plus grands robots sont normalement équipés. Cet article contribue donc au mémoire en présentant un

observateur de force externe qui utilisent les capteurs disponibles sur notre plate-forme robotique, le Nao. Plus spécifiquement, la contribution principale de ce travail est : (1) un algorithme d'estimation de force externe appliquée sur un robot humanoïde qui utilise un ensemble limité de capteurs et qui est donc applicable au robot de petite taille.

Résumé français : Dans cet article, on présente une méthode permettant d'estimer la grandeur d'une force externe appliquée sur un robot humanoïde. L'approche présentée ne requiert pas l'utilisation de capteurs de force/couple, mais utilise plutôt des capteurs de pression tactile sous les pieds du robot. L'approche est particulièrement intéressante pour les humanoïdes de petite taille tels que le Nao ou Darwin-OP. L'idée principale est de modéliser le système par une version linéaire d'un pendule inversée auquel on applique une force, et de comparer la réponse du modèle avec le robot en utilisant la centrale inertielle et les capteurs de pression sous les pieds du robot. L'algorithme a été validé sur un robot humanoïde Nao pour estimer des forces appliquées dans plan sagittal à l'aide de deux types d'expérimentations. Les résultats obtenus valident l'efficacité de la méthode.

3.1 Abstract

In this paper, we introduce a method to estimate the magnitude of an external force applied on a humanoid robot. The approach does not require using force/torque sensors but instead uses measurements from commonly available force-sensing resistors (FSR) inserted under the feet of the humanoid robot. This approach is particularly interesting for affordable medium-sized humanoid robots such as Nao and Darwin-OP. The main idea is to use a simplified dynamic model of a linear inverted pendulum model (LIPM) subjected to an external force, and the information from the robot inertial measurement unit (IMU) and FSR sensors.

The proposed method was validated on a Nao humanoid robot to estimate the external force applied in the sagittal plane through two experimental scenarios, and the results pointed out the efficiency of the proposed observer.

3.2 Introduction

Humanoid robots are good candidates to perform manipulation and transport tasks since they possess articulated arms. These tasks require the robot to be able to adjust its gait in order to take into account the external forces exerted on it. In such situations, the robot usually uses his interoceptive/exteroceptive sensors to estimate those forces and a representative dynamic model to generate stable patterns.

A situation in which the knowledge of the transported mass or applied force could be very useful to generate a more stable gait is the transportation of an object on a cart by a robot. This scenario is considered in [Rioux et Suleiman, 2015], where a motion planner uses different sets of motion primitives depending on the mass transported on the cart. The estimation of the load is done by making the robot execute a turning in place motion and by looking at the error between the planned motion and the actual robot position. Although this approach can effectively differentiate a light load from a heavy one, the differentiation is done by roughly applying a binary operator to the error. The planning algorithm then chooses the heavy load or small load primitives set accordingly. A better estimation of the load could allow a more optimal motion planning as more primitive sets adapted to different load could be used. Moreover, by integrating the estimated external force into the pattern generation module, more stable motions can be obtained even with a relatively heavy load.

3.2.1 Relevant works

Improving the robustness of humanoids walk against disturbance is a topic of interest since these robots are expected to perform tasks in a variety of human environments. In[Kaneko *et coll.*, 2012], a disturbance observer that estimates the magnitude of an external force is presented. The observer uses measurement from an IMU and six-axis force/torque (F/T) sensors located at the ankles to detect and estimate the magnitude of a strong force such as a kick to the chest of the robot or a collision with the environment. Whereas a kick to the chest can be represented by an impulse input applied to the system, a pushing motion is equivalent to a step input and it is therefore expected that the observer would not have the same performance in the latter case.

Some research work were also conducted to estimate an external force applied on a small humanoid robot. In [Berger *et coll.*, 2014] and [Berger *et coll.*, 2015], the authors are interested in predicting the perturbations transmitted to a robot in a human-robot interaction. Essentially, the proposed approach is to generate a probabilistic model of the sensors output to predict future readings. Then, if the measurements are not in accordance with the model, it can be concluded that an external force is applied. The authors were able to use the system in a human-robot interaction to infer the human intention and move the robot accordingly. However, in a robot-human interaction, the robot does not need to modify its walking gait since the human will apply more or less force depending on the robot reaction. But if the humanoid is pushing or transporting an object, it must adjust its gait to remain stable. In this case, an estimation of the force magnitude would be necessary.

In [Harada *et coll.*, 2003a,b], the ZMP dynamic is analysed for a humanoid robot performing a pushing manipulation task. In this case, the external force is directly measured through F/T sensors located in the wrist of a HRP-2 robot. Their results reveal that the robot falls if the motion generator does not consider the exerted force on the grippers. On the other hand, the required compensation can be extracted by computing two ZMPs. The first ZMP, referred as the generalized ZMP, is computed by considering the gravity and the reaction force and moment on the floor. A second ZMP is computed by considering the external force applied on the grippers. The difference between those two ZMPs is added to the desired ZMP trajectory, which is generated using the LIPM model. This procedure was successfully applied in simulation to generate a stable motion. There are also numerous works[Sato et Ohnishi, 2008][Prahlad *et coll.*, 2008] where the ZMP position error is monitored and integrated into the control law. In these cases, quantifying the perturbation and determining its origin were not addressed. As the ZMP was

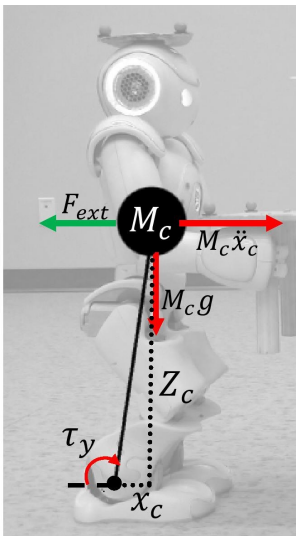


Figure 3.1 Simplified model of a humanoid robot with an external force

successfully used in all these previous work to make humanoid walk more robustly against disturbance, a plausible approach to approximate an external force would be monitoring the ZMP variation and linking it to an external force. The objective is then to have a dynamic model of the task being executed and to be able to measure the actual ZMP with enough precision.

In this work, assuming the robot is equipped with force-sensing resistors (FSR) under the feet, we propose a method for estimating the magnitude and direction of an external force in the sagittal plane using the LIPM model and ZMP measurements using the FSR. The proposed observer was designed for and validated on a Nao humanoid robot. Nao [Gouaillier *et coll.*, 2009] is a medium-sized humanoid robot manufactured by Aldebaran. On the contrary of complex and highly sophisticated humanoid robots, such as HRP-2 or Atlas, Nao does not have six-axis F/T sensors and possesses only FSR under the feet.

The LIPM model and the dynamic equations with an external force are presented in Section 3.3. Section 3.4 deals with estimating the ZMP using the available sensors. The external force-observer architecture and implementation is addressed in section 3.5. Finally, in section 4.5, experimental results are presented and the observer performance is analyzed and discussed.

3.3 Dynamic models

3.3.1 LIPM dynamic

LIPM model has been widely used to generate stable walking patterns [Kajita et Tani, 1995]. According to this model, the Center of Mass (CoM) only moves under the action of the gravity. The dynamics of the LIPM can be decoupled within each axis and therefore we only show hereafter the equation in the sagittal plane. The motion dynamic can be written as

$$M_c \, \ddot{x}_c = \frac{M_c \, g \, x_c}{Z_c} \tag{3.1}$$

where M_c is the mass of inverted pendulum, Z_c is the height of CoM, g is the magnitude of gravity acceleration, x_c and $\ddot{x}_c$ are the position and the acceleration of the projection of CoM on the sagittal axis. Note that x is expressed in the pivot frame, which corresponds to the ankle on the real system.

3.3.2 Dynamic model with external force

The general dynamic model of a robot walking with an external force applied is presented in Fig. 3.1, and is defined as follows :

$$M_c \, \ddot{x}_c = \frac{M_c \; g \, x_c}{Z_c} - F_{ext} \tag{3.2}$$

where F_{ext} is an external force. For instance, this force might be the result of the robot pushing/lifting an object or interacting/collaborating with a human.

Here, we have a system identification problem where we need to estimate F_{ext} in order to generate a stable motion. Although monitoring $\ddot{x}_c$ or x_c might give us an insight into the external force that is applied, it would not be very useful in the case of a position-controlled robot such as Nao. Unless the external force is strong enough that the motors of the robot are unable to keep their position, the CoM position will not be affected. However, it is worth mentioning that a strong punctual disturbance such as a push will affect the CoM position and acceleration as it is the main idea behind the external force observer in [Kaneko *et coll.*, 2012].

ZMP without external force (x'_{ZMP})

First, we consider the ZMP according to its general definition for a humanoid robot, approximated by a LIPM, without external force. In other words, we only consider

the gravity and the inertial force of the robot. Recalling that the ZMP is a point on the ground where the inertial and gravity moments cancel out, it is defined as

$$x'_{ZMP} = x_c - \frac{Z_c \ddot{x}_c}{g} \tag{3.3}$$

We will refer to the ZMP computed without considering the external force as x'_{ZMP}. x'_{ZMP} can be directly obtained from the desired ZMP trajectory. It can also be computed using the acceleration from the IMU to approximate $\ddot{x}_c$ and the direct kinematic to get x_c.

ZMP

The true ZMP can be found by considering every force acting on the system. From (4.1), the ZMP is defined as :

$$x_{ZMP} = x_c - \frac{Z_c \ddot{x}_c}{g} - \frac{F_{ext} Z_c}{M_c g} \tag{3.4}$$

We will refer to the ZMP computed considering the external force as x_{ZMP}. The basic idea of our external force observer is that the difference between the actual ZMP (x_{ZMP}) and the planned ZMP (x'_{ZMP}) is proportional to the external force. The main challenge is then to get a good estimate of the actual ZMP. The next section deals with estimating the ZMP with measurements from the available sensors.

3.4 ZMP Estimation

As presented in [Shim *et coll.*, 2004], the ZMP can be estimated with force-sensing resistors located under the feet. The main idea is that by measuring the applied force on each force sensor, it is possible to estimate the center of pressure (CoP) by calculating the position of the equivalent force. Then, recalling that the ZMP and the CoP are the same point if the robot is in a stable configuration [Vukobratović et Borovac, 2004], we can estimate the ZMP by computing the CoP

$$x_{ZMP} = x_{CoP} \tag{3.5}$$

3.4.1 Center of pressure measurements

Each foot of Nao possesses 4 FSR that each returns a force between 0 and 25N. .Using these measurements, one can easily compute the center of pressure on one foot as

$$x_{CoP} = \frac{\sum_{i=1}^{4} F_i \, x_i}{\sum_{i=1}^{4} F_i} = \frac{\sum_{i=1}^{4} F_i \, x_i}{F_T} \tag{3.6}$$

where F_i is the force measured at the i^{th} FSR, x_i is the position of the i^{th} FSR in the foot frame and F_T is the total force applied on the foot.

During walking, we assume that the robot is always in a single support phase. Therefore, the support foot must be detected to apply the previous formula. A possible solution to determine the support foot is to consider the total force applied on each foot (measured with the FSR) as the decision variable. However, using a simple boolean operator results into false support foot detection since the force measured by the FSR becomes highly noisy when a foot lands on the floor.

The adopted solution, similar to [Xinjilefu *et coll.*, 2015], is to implement a Schmitt trigger to process the support foot state. The Schmitt trigger uses a low and a high threshold to determine the value of the output. In our case, the input to the trigger is the total force on the right foot minus the total force on the left foot. Hence, positive value means that the support foot is the right and a zero value means the support foot is the left. Satisfactory results were obtained by setting the positive-going threshold at $2N$ and the negative-going threshold at $-2N$.

However, when the robot is immobile and in double-support mode, the ZMP (or CoP) can be simply found using the following formula

$$x_{CoP} = \frac{x^l_{CoP} \, F^l_T + x^r_{CoP} \, F^r_T}{F^l_T + F^r_T} \tag{3.7}$$

where x^*_{CoP} and F^*_T are computed as in (3.6) for the left and right feet.

3.4.2 External force observer

From (3.3) and (3.4) , the external force F_{ext} can be easily found

$$F_{ext} = M_c \, g \, \frac{(x'_{ZMP} - x_{ZMP})}{Z_c} \tag{3.8}$$

where x'_{ZMP} is the ZMP computed using the LIPM model and x_{ZMP} is measured with the FSR.

3.5 Implementation

The architecture of the observer is summarized in Fig. 3.2. Implementation details are presented in the following subsections.

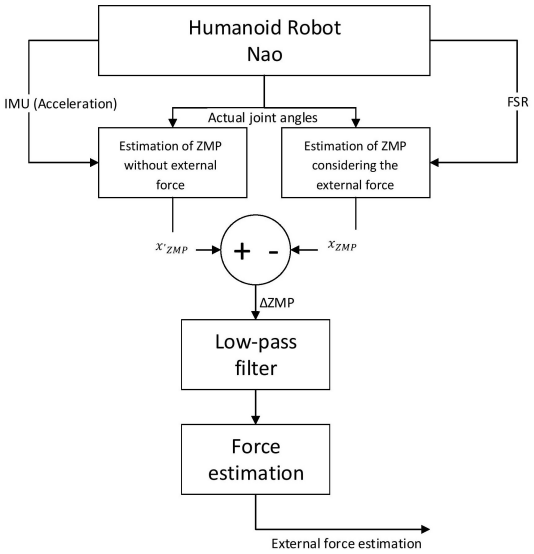


Figure 3.2 Block diagram of the observer

3.5.1 x'_{ZMP} Computation

As mentionned before, x'_{ZMP} corresponds to the position at which the ZMP would be if no external force is applied. For humanoid robots that use a ZMP-based control scheme to generate the walking gait, it typically corresponds to the desired ZMP trajectory. However, to compute the ZMP with (3.3) the position and acceleration of the CoM must be approximated.

CoM acceleration

The acceleration of the CoM can be extracted from the IMU, which is a standard part of a humanoid robot sensors. As for Nao, the IMU provides measurements from a three-axis accelerometer and a two-axis gyroscope. Also, an existing on-board algorithm provides an estimation of the torso orientation. In this work, we make the assumption that the CoM coincides with the IMU. Therefore, incoming data from the accelerometer can be used to approximate the acceleration of the CoM. Fig. 3.3 presents a block diagram of

the raw acceleration data processing. The projection into an inertial frame and the gravity compensation steps are implemented as presented in [Bellaccini *et coll.*, 2014].

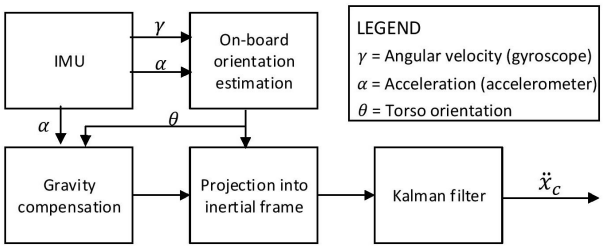


Figure 3.3 Block diagram of the processing done on the raw acceleration data

CoM Position

In order to determine the CoM position, we make the assumption that the trunk of Nao is the center of mass. This assumption is also made in the built-in walk engine of Nao as presented in [Gouaillier *et coll.*, 2010]. Therefore, simple direct kinematic computation with encoder readings from the servo motors is used to determine the position of the CoM in the support foot frame. The support foot is determined using the method presented in Section 3.4.1.

3.5.2 Low-pass filtering (Sway motion cancellation)

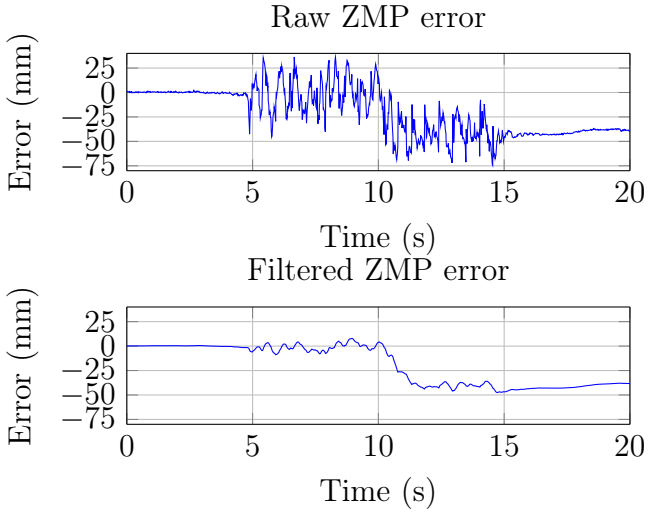


Figure 3.4 ZMP error during a sagittal walk before and after filtering. At $T = 10s$, a constant force was applied

During the walk, the torso of a humanoid robot will naturally oscillate in what is typically called the sway motion. Obviously, the ZMP will also oscillate in the same

manner. If this swaying oscillation is not removed at some point, it will highly degrade the performance of the observer. This problem has already been tackled in [Oriolo *et coll.*, 2015], where the filtering of the Nao robot sway motion is analysed. It is mentionned that the frequency of the sway motion for the Nao robot is close to 1 Hz. This claim was validated on our robot, and a first order low-pass filter with a cut-off frequency of 0.6 Hz has been used. Fig. 3.4 presents the error signal between the two ZMPs before and after filtering during a sagittal walk.

3.6 Experimental Results

The observer was tested in two scenarios : stationary robot with an external force and a walking robot with an external force. All the experiments results were generated with the following parameters : $g = 9.81\ m/s^2$, $Z_c = 0.315$m, sampling period $(T) = 16.7$ ms and $M_c = 4.5$Kg.

3.6.1 Case Study 1 : Stationary with external force

The dynamic model of the first scenario considered is presented in Fig. 3.6. The used setup for this experiment is presented in Fig. 3.5. In this experiment, the robot is standing still and attached from the waist to a mass in the form of a water bottle. At some point, the mass is released and the force propagates to the robot through a basic pulley system.

Figure 3.5 Experimental setup used to apply a known external force

Results

Fig. 3.8 presents the results of an experiment where a constant force of approximately 3.8N was applied on a static robot. Fig. 5.11(a) shows that the external force has no significant effect on the CoM acceleration and that the servo motors are able to keep their position despite the added force since the x'_{ZMP} is not affected. On the other hand,

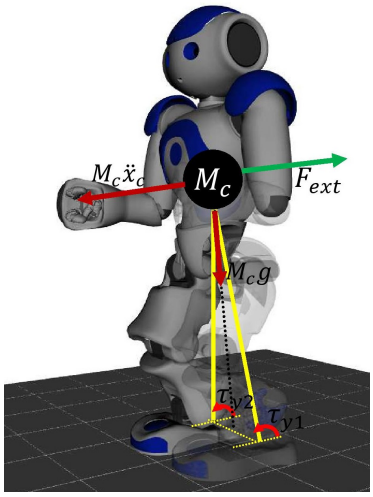


Figure 3.6 Case study 1 : stationary robot with an external force

the ZMP measured with the FSR is rapidly shifted by more than 3 cm. In Fig. 5.11(b), the difference between the ZMPs is presented after low-pass filtering. Since the robot is not walking, the filter presented in Section 3.5.2 is not necessary here. Instead, a low-pass filter with a cut-off frequency of 4 Hz was used to smooth the signal. Fig. 3.8(c) shows that the observer estimated force is close to the actual one. In order to characterize the performance of the observer, we define the settling time as the time needed for the output to reach and stay within a 20% margin of the reference value. In this case, the settling time is close to 0.5s. Similar results were obtained in experiments performed with external force ranging from 1N to 4N. In each case, the observer was able to estimate the external force within a 20% error margin.

3.6.2 Case Study 2 : Walking with external force

The experimental setup used in this case is the same as in Case 1 (Fig. 3.5). Initially, the robot is standing still and attached from the waist to a mass which is in full contact with the ground. The robot then starts to walk forward and at some point, the rope is completely tensed and the mass is lifted off the ground by the walking robot, as shown in Fig. 3.7. The lifted mass acts as an external force pulling the robot backwards.

Results

The experimental results of a forward walk with an external force of 3N is presented in Fig. 3.9. As shown in Fig.5.12(a), the two ZMPs are similar until the external force is applied at around 10s. At this point, the x_{ZMP} is shifted to the back as opposed to the x'_{ZMP} that continues to oscillate around 0. However, as demonstrated in Fig. 5.12(b),

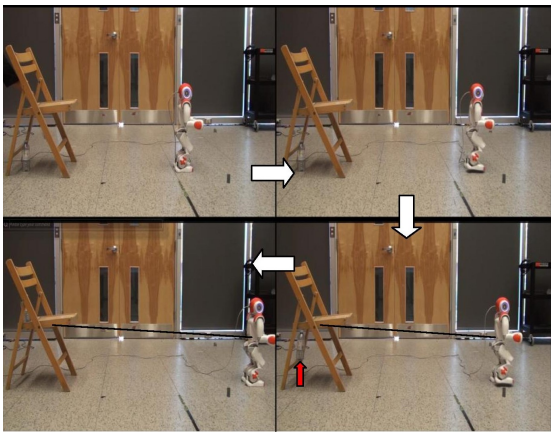


Figure 3.7 Snapshots of an experiment where a mass is lifted off the floor by a Nao robot walking forward

there is a small error between the two ZMPs even when no force is applied. Thus, a threshold operation is applied on the error signal to avoid detecting a false external force. Accordingly, the observer output is more stable but external forces of magnitude less than 0.5N are not detected. In Fig. 3.8(c), the observer estimated force is given along with the true external force. One can figure out that the observer successfully estimated the true external force. Also, as might be seen, the settling time is slightly more than 1s. In this experiment, it corresponds to two walking steps for the robot.

During similar experiments, the observer was able to estimate an external force ranging from 0.8N to 3.8N within a 20% error margin. Note that the maximum force that we could apply on the walking Nao, using the robot built-in walk engine, without making it fall, was 3.8N.

3.7 Conclusion

In this paper, we introduced a method to estimate the magnitude of an external force acting on a humanoid robot without using expensive 6-axis force/torque sensors. Essentially, it uses measurement from force-sensing resistors located under the feet of the robot to estimate the position of the ZMP and compare it to a reference ZMP that is computed using the linear inverted pendulum model. This approach is mainly interesting for medium-sized humanoid robots, and it was successfully validated on a Nao robot in two different scenarios.

Future work will focus on integrating the external force observer into the pattern generation module.

Acknowledgment

This research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

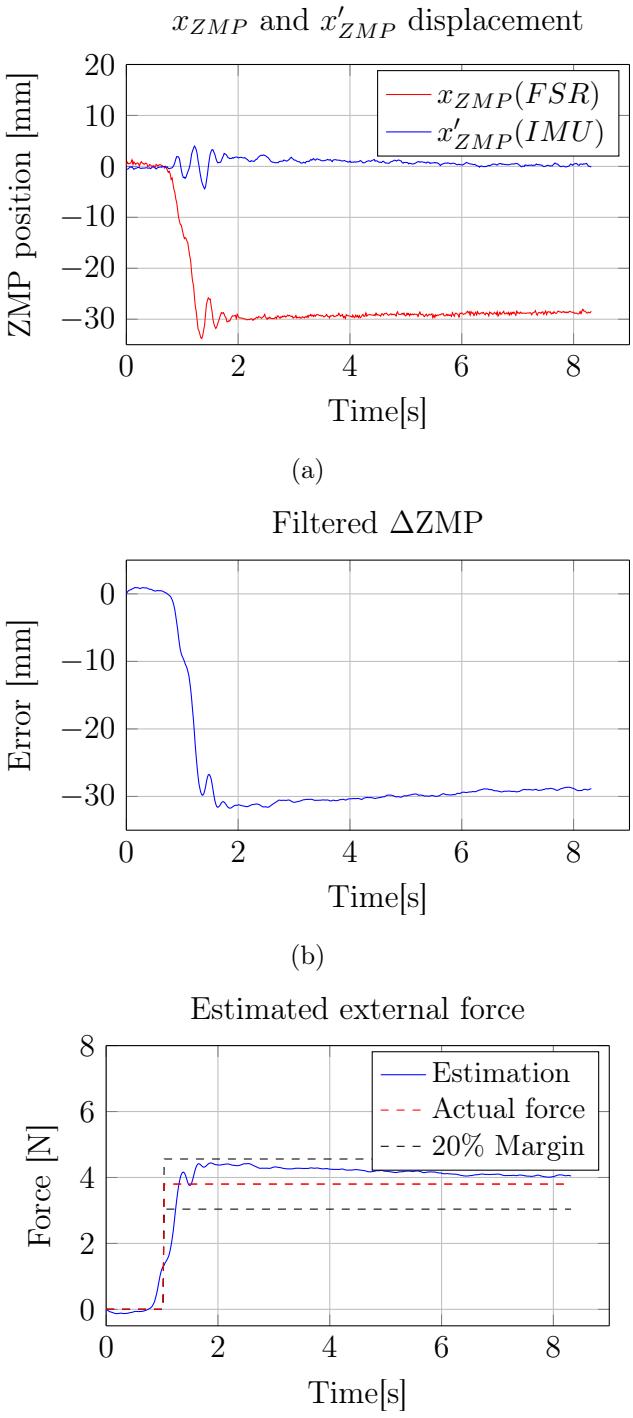
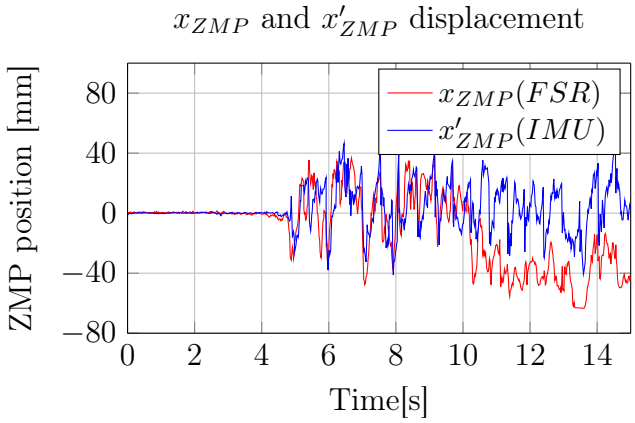
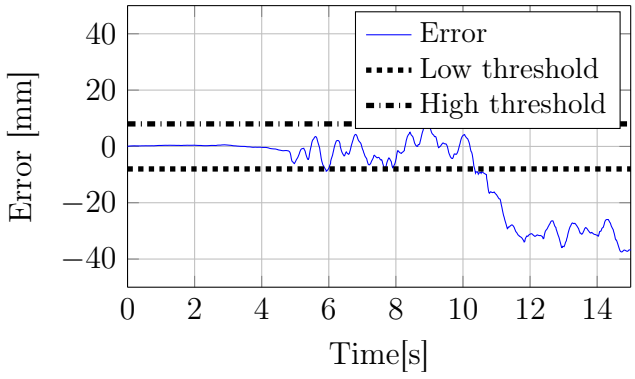


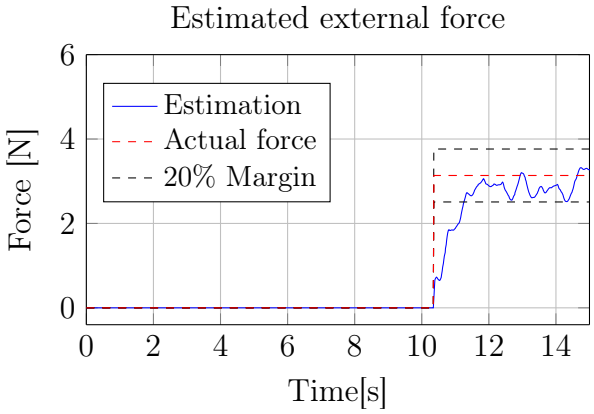
Figure 3.8 **Case Study 1** : ZMP variation and force estimation when an external force of approximately 3.8N is pulling a stationary Nao backward. At approximately 1 second, the force is applied. (a) The displacement of x_{ZMP} and x'_{ZMP} when the force is applied (b) The difference between the two ZMPs after applying a low-pass filter (c) The actual external force and the observer estimation



(a)



(b)



(c)

Figure 3.9 **Case Study 2** : ZMP variation and force estimation when an external force of approximately 3N is applied on a walking Nao. At approximately 10 seconds, the force is applied. (a) The displacement of x_{ZMP} and x'_{ZMP} when the force is applied (b) The difference between the two ZMPs after applying a low-pass filter (c) The actual external force and the observer estimation

CHAPITRE 4

Stratégie de contrôle pour un robot humanoïde transportant une charge à l'aide d'un chariot

Auteurs et affiliation :

L.Hawley : étudiant à la maîtrise en génie électrique, Université de Sherbrooke, Faculté de génie, Département de génie électrique et informatique

W.Suleiman : Professeur, Laboratoire de robotique intelligente, interactive, intégrée et interdisciplinaire (IntRoLab), Université de Sherbrooke, Faculté de génie, Département de génie électrique et de génie informatique

État de la publication : Version finale publiée

Date de publication : Septembre 2017

Référence : Hawley, L., & Suleiman, W. (2017). Control Strategy and Implementation for a Humanoid Robot Pushing a Heavy Load on a Rolling Cart. Dans IEEE International Conference on Intelligent Robots and Systems (IROS).

Conférence : 2017 IEEE International Conference on Intelligent Robots and Systems (IROS)

Titre français : Stratégie de contrôle et implémentation pour un robot humanoïde déplaçant une charge lourde en utilisant un chariot

Contribution au document : La seconde partie du projet de recherche fût de valider qu'il était possible d'utiliser l'estimateur de force externe présenter au chapitre 3 pour contrôler de façon stable un robot humanoïde effectuant une tâche de transport. Toutefois, au lieu de travailler directement sur la tâche de transport collaboratif, le

transport d'objet à l'aide d'un chariot à roues à d'abord été analysé. Cela a permis de valider et de caractériser la performance de l'approche dans un environnement plus simple que le transport collaboratif où la synchronisation et l'échange d'informations entre les robots doivent également être gérées. Ainsi, cet article contribue au mémoire en présentant une stratégie de contrôle permettant à des petits humanoïdes d'augmenter leurs capacités en terme de transport d'objet en plus de définir l'approche générale qui sera également utilisée pour le transport collaboratif d'une charge.

Résumé français : L'article introduit une stratégie de contrôle permettant de générer une marche stable pour un robot humanoïde transportant une charge importante à l'aide d'un chariot. Comparativement à certaines approches qui requièrent des capteurs de force/couple pour mesurer l'interaction entre le robot et la charge à pousser, on présente une approche basée modèle (*model-based*) qui peut être implémenté sur la plupart des robots en raison de la faible charge de calcul requise. Tous les aspects du design sont traités en partant de la formulation et la validation du modèle dynamique jusqu'à l'implémentation et la validation sur un véritable robot humanoïde. Les résultats expérimentaux montrent que le contrôleur est en mesure de stabiliser un robot Nao durant une tâche de transport durant laquelle le robot transporte l'équivalent de son propre poids à l'aide d'un chariot.

4.1 Abstract

In this paper, we introduce a control strategy aimed at generating a stable walking pattern for a humanoid pushing a heavy load on a cart. In contrast to previous approaches that rely on force/torque sensors to measure the interaction between the robot and the pushed object, we present a simple model-based controller that can be implemented on most robots due to its computationally efficient design. Every aspect of the controller design is covered, from the formulation and validation of the dynamic model, to the implementation and validation on an actual humanoid robot. The experimental results show that the controller can efficiently make a NAO humanoid transport, in a stable way, the equivalent of its own weight on a rolling cart.

4.2 Introduction

In order to work as efficiently as possible, robots should be able to make use of their environment or available tools. In the case of humanoid robots, one of their biggest strength is their high mobility due to the biped structure. However, it is also a weakness as they are usually unable to carry heavy load without being destabilized contrarily to a wheeled-robot that remains stable. A potential solution to this problem is simply using mobile carts to transport massive or heavy object. However, in order to use such device efficiently, the humanoid robots navigation schemes and control algorithms should be accordingly modified to take the cart into consideration.

In our previous work [Rioux et Suleiman, 2015], we presented a navigation approach for a humanoid robot pushing a heavy object on a rolling cart in a cluttered environment. This approach could effectively be used to transport a load that the humanoid robot would not have been able to carry otherwise. However, the dynamic of the system was not analyzed and no corrections were applied on the walking gait of the robot to compensate for the added mass transported by the robot. The focus of this research is to analyze the impact of the cart and transported load on the humanoid stability and come up with appropriate and computationally efficient solutions to control the whole system.

4.2.1 State of the art

A controller architecture for pushing an object by humanoids is presented in [Harada *et coll.*, 2003a]. In that work, the authors analyzed the dynamic of a humanoid pushing an object with the arms and assessed that the desired Zero Moment Point (ZMP) trajectory must be modified to keep the robot stable despite the external force generated

by the object weight and the friction with the floor. The required ZMP shift is defined as a function of the external force detected by the robot. Thus, the dynamic behaviour of the pushed object is not considered as it is only approximated as a constant external force. Moreover, the authors do not point out how the external force, resulting from pushing the object, should be measured or approximated.

In a more recent work [Harada *et coll.*, 2007], the same authors implemented a controller on a HRP-2 robot pushing a table. The proposed approach is to only push the object while the robot is in a double-support phase. While in double-support phase, a force-controlled loop is driving the arm of the robot in order to apply a desired force on the object. Also, the reference ZMP position is modified to ensure that the robot remains stable considering the reference force applied by the arms on the object. Although this approach is efficient for pushing a heavy object on the floor, a significant drawback is that the force-controlled loop requires accurate measurements of the force applied by the robot's hands. Those forces are measured using force/torque sensors in the wrists of the humanoid robot and these sensors are not available on all robot, e.g. medium-sized humanoid robots such as Nao or Darwin-OP.

Using wheeled platforms to help humanoids transport objects has also received some attention in the literature as it is a skill that could be useful in both a manufacturing or personal robotics context. For example, in [Nozawa *et coll.*, 2012], a HRP-2 robot is used to push a variety of objects ranging from a wheelchair to a simple holonomic transport cart. The authors assessed the importance of integrating the force generated by the pushed object in the walk controller to ensure stability. The proposed system is based on an online friction estimator [Nozawa *et coll.*, 2011a] that is able to estimate the friction forces generated by the object using measurements from force/torque sensor located in the wrists

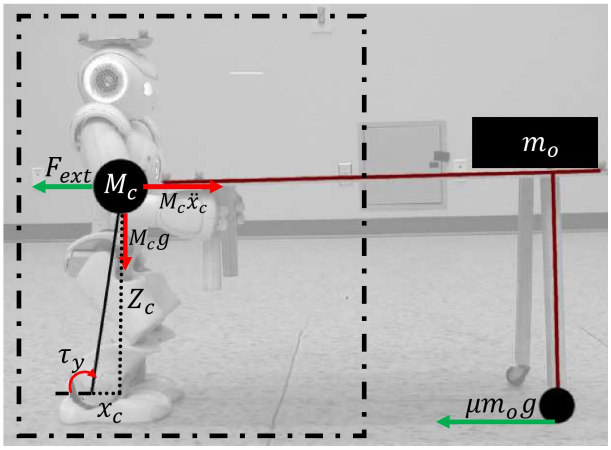


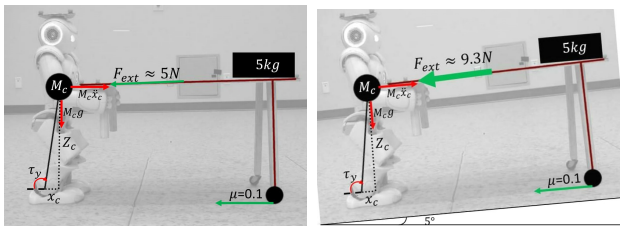
Figure 4.1 Humanoid robot pushing a load on a rolling cart

and ankles of the robot. Then, the desired ZMP trajectory is modified and supplied to the preview controller to consider the external force. Again, a disadvantage of this approach is that it relies heavily on the good performance of the force/torque sensors. Moreover, according to their approach, the robot must be controlled in force in order to estimate the friction force. However, a lot of humanoid robots are only position-controlled and do not possess such sensors, it would be therefore interesting to have another method that does not rely too heavily on the measurement of those interaction forces.

In most research related to pushing objects by humanoid robots, modelling the external forces is not really treated as those forces can be reliably and directly measured with the available sensors. However, there are multiples advantages of having a model of the force generated by the object as this knowledge can be useful to predict how that force will vary in different conditions.

4.2.2 Contribution

In this work, we present a different control strategy for a humanoid robot pushing a heavy weight on a mobile-cart. The proposed method enhances the simple Linear-Inverted Pendulum Model (LIPM) of a humanoid robot to also include the dynamic model of the cart-load system. Contrarily to the other previously presented methods, our procedure can be used by position-controlled robots that are not equipped with expensive force/torque sensors. In addition, our approach could predict the external force generated by the rolling-cart system as the environment changes (ex : change in floor type or presence of a slope) as illustrate in Fig. 4.2.



(a) Flat (b) 5^o slope

Figure 4.2 The external force generated by the cart and mass on a flat floor and on a small slope. In this case, the force exerted on the robot is nearly doubled in the case of slope compared to a flat floor. A model of the external force can be used to predict such variation and give an idea if the robot can push the cart up the slope or if it should find another path.

The article is organized as follows. The dynamic model of the cart-pushing task and the identification of the cart parameters are presented in Section 4.3. Section 5.7

tackles the problem of maintaining the robot stability while executing the transportation task with the cart. Finally, the results of our experiments carried out on a Nao humanoid are presented in section 4.5.

4.3 Dynamic models

4.3.1 LIPM dynamic with an external force

In this work, the LIPM is used to generate a stable walking pattern [Kajita et Tani, 1995]. In the case of an external force in the horizontal plane, $F_{ext} = [F_x\, F_y\, 0]^T$, the motion dynamic can be written as follows :

$$\begin{aligned} M_c\, \ddot{x}_c &= \frac{M_c\; g}{Z_c}\, x_c + \frac{1}{Z_c}\tau_y + F_x \\ M_c\, \ddot{y}_c &= \frac{M_c\; g}{Z_c}\, y_c - \frac{1}{Z_c}\tau_x + F_y \end{aligned} \tag{4.1}$$

where M_c is the mass of inverted pendulum, Z_c is the height of CoM, g is the magnitude of gravity acceleration, x_c, $\ddot{x}_c$, y_c and $\ddot{y}_c$and are respectively the position and the acceleration of the projection of CoM on the x and y axis. τ_x and τ_y are respectively the torques around x and y axis.

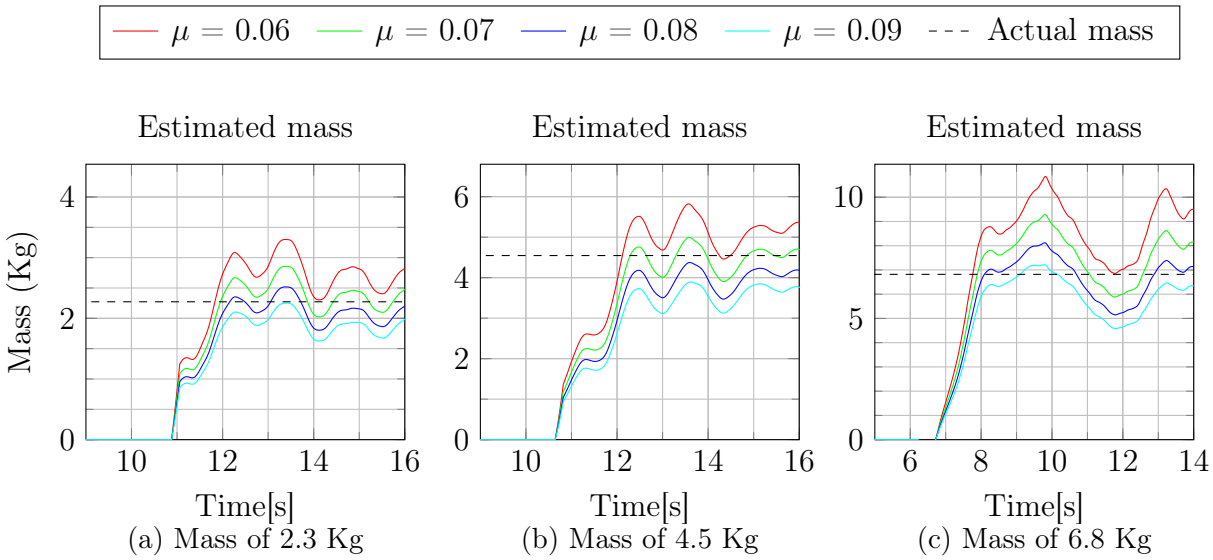
In a previous work [Hawley et Suleiman, 2016], we proposed a method to estimate an external force applied in the horizontal plane. This method is mainly designed for small humanoid robots, such as Nao robot, and only uses the information from Force Sensitive Resistors (FSR) and Inertial Measurement Unit (IMU).

4.3.2 Cart-pushing task dynamic

The dynamic model of the cart-pushing task presented in Fig. 4.1, in which the robot is moving forward in the sagittal plane ($x-$axis), is defined as follows :

$$\begin{aligned} M_c\, \ddot{x}_c &= \frac{M_c\, g}{Z_c}\, x_c + \frac{1}{Z_c}\tau_y - F_{ext} \\ M_c\, \ddot{y}_c &= \frac{M_c\, g}{Z_c}\, y_c - \frac{1}{Z_c}\tau_x \\ F_{ext} &= m_o\, \ddot{x}_c + \mu\, m_o\, g \end{aligned} \tag{4.2}$$

where μ is the coefficient of rolling friction and m_o is the mass of the load on the cart. We suppose that the mass is placed above the cart legs.



(a) Mass of 2.3 Kg (b) Mass of 4.5 Kg (c) Mass of 6.8 Kg

Figure 4.3 Estimation of the transported mass for three experiments using Eq. (4.3) and varying rolling friction coefficient μ

From the previous equations, one can see that the origin of the external force can be decoupled into two terms, the first term being the result of the added mass and the second one by the rolling friction on the wheels. The rolling friction coefficient is a function of multiple parameters that are intrinsic to the cart, mainly the wheels diameter and hardness, and also to some other external parameters such as the floor type and surface condition. From Eq.(4.2), provide that the rolling friction coefficient is known, the mass carried on the cart can be easily estimated by measuring the external force and the acceleration as follows :

$$m_o = \frac{F_{ext}}{\mu\, g + \ddot{x}_c} \tag{4.3}$$

The following section provides a simple way to estimate the rolling friction coefficient of a cart.

4.3.3 Cart Parameter Identification

In order to generate a stable walking pattern while pushing the cart, it is possible to measure the force generated by the cart and then directly incorporate this force in the stability control loop of the robot. However, as above-mentioned, having an estimation of the transported load can sometimes be really important in the motion planning phase. For example, while a heavy load can be relatively easy to transport with a cart on a hard

and flat floor, it can be much more difficult if the robot pushes the cart on a carpet or on a slope. Such knowledge can be used by the path planner to prevent from accessing zones that will likely destabilize the robot or to optimize the energy consumption of the robot by avoiding high-friction surfaces or steep slopes.

A simple way to measure the friction coefficient of an object is presented in [Nozawa *et coll.*, 2011b], where the authors slowly increase the force applied by the humanoid robot arms on the object. At some point, the object starts to move, and thus the measured force stabilizes at a given value. If the mass of the object m_o is known, the friction coefficient μ can then easily be found with $F = \mu m_o g$. However, it is not possible to reproduce this procedure with a Nao robot since the motor can only be controlled in position.

In order to estimate the parameters of the cart, three experiments were performed with a Nao robot pushing different masses using a rudimentary wheeled-table. In each experiment, we estimated the external force applied on the robot using our force observer [Hawley et Suleiman, 2016], and tried to correctly guess the mass transported on the cart by using Eq. (4.3) and varying the cart-parameter μ.

The results of those experiments along with the mass estimation using different μ values are presented in Fig. 4.3. Note that in this case, the parameter μ combines both the intrinsic cart-parameter, i.e. wheel diameter, and environmental parameters, i.e. floor type, since our laboratory has a uniform floor. Otherwise, the experiments would have been conducted on the different available surfaces and the coefficient would have been split into two terms. A quick look at the results shows that a μ value of about 0.07 yields a reasonable estimate for the three masses, and this value is therefore used in the sequel of the paper. The following section takes a look at the effect of the carried mass on the robot ZMP in simulation as well as on a real Nao robot.

4.3.4 Model Validation

Using the dynamic model presented in section 4.3.2, the ZMP dynamic was simulated in Matlab for a scenario where a humanoid robot pushes the previously characterized cart with a mass of 5kg. Recalling that the ZMP is the point on the floor where

the sum of the moment is null, it is expressed as :

$$\begin{aligned} x_{ZMP} &= x_c - \frac{Z_c}{g}\,\ddot{x}_c - \frac{Z_c}{M_c\,g}\,F_{ext} \\ x_{ZMP} &= x_c - \frac{Z_c}{g}\,\ddot{x}_c - \frac{m_o\,Z_c}{M_c\,g}\,\ddot{x}_c - \frac{\mu\,m_o\,Z_c}{M_c} \\ x_{ZMP} &= x_c - \frac{(m_o + M_c)\,Z_c}{M_c\,g}\,\ddot{x}_c - \frac{\mu\,m_o\,Z_c}{M_c} \\ y_{ZMP} &= y_c - \frac{Z_c}{g}\,\ddot{y}_c \end{aligned} \tag{4.4}$$

Where x_{ZMP} and y_{ZMP} are the projections of the ZMP on the floor ($x-y$ plane). Fig. 4.4(a) and Fig. 4.4(b) respectively presents the planned and actual global ZMP in the sagittal (x_{ZMP}) and coronal (y_{ZMP}) plane of the robot as the mass is added on the cart at $T \approx 4\,s$. The local x_{ZMP} (expressed in the support foot ankle frame) is also presented in Fig. 4.4(c) to clearly show the effect of the cart and mass on the ZMP dynamic. First, the rolling friction caused by the cart wheels generates a constant shift of the ZMP toward the back. Second, the mass carried on the cart generates a dynamic force as a result of the robot center of mass acceleration. Fig. 4.4(b) validates that in the case of a forward push, the y_{ZMP} is not affected.

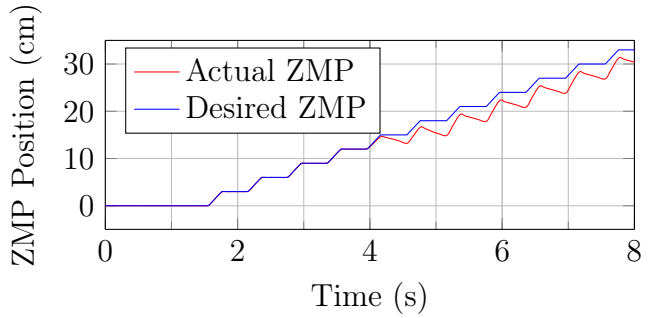
The experiment was then carried out on a real Nao robot to validate the simulation results. Fig. 4.5 presents the local ZMP position during the conducted experiment. At first the NAO humanoid is pushing an empty cart. Then, at $T \approx 22s$, a mass of 5 kg was added on the cart. Fig. 4.5 points out that the ZMP shift is similar to what was found in simulation as the ZMP moves approximately 2 cm backward whereas a shift of approximately 2.4 cm was found in simulation.

4.4 Control law

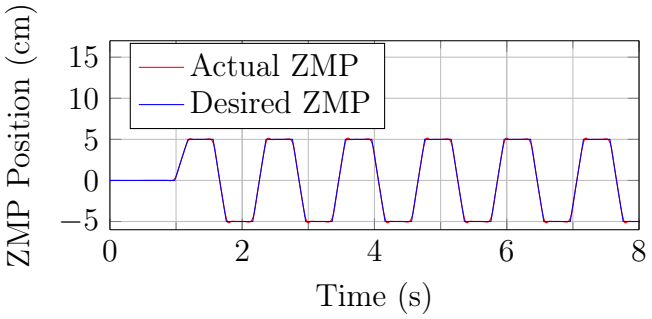
4.4.1 ZMP Preview Control

Now that we have a valid model for the external force generated by the rolling-cart and the transported mass, a control law considering the whole dynamic must be developed.

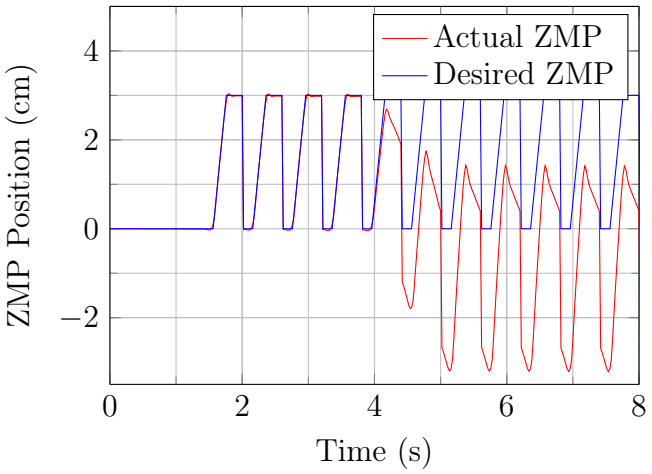
To this end, we designed an optimal preview servo controller following the method defined in [Kajita *et coll.*, 2003].



(a) Global ZMP in the sagittal plane (x-axis)



(b) Global ZMP in the coronal plane (y-axis)



(c) Local ZMP in the sagittal plane (x-axis)

Figure 4.4 Simulation result for a scenario where a mass of 5 kg is added on a cart pushed by a Nao humanoid. The ZMP in the coronal plane (b) of the robot is not affected in this case contrarily to the ZMP in the sagittal plane((a) and (c)).

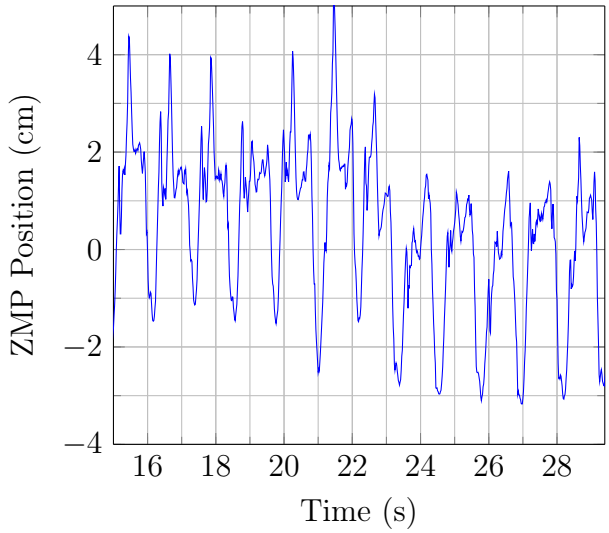


Figure 4.5 The local ZMP measured during an experiment. At around $T = 22s$, a mass of 4Kg was added on the cart

Let us first define a new variable p such that :

$$p(t) = \begin{bmatrix} x_{ZMP} + \frac{\mu m_o Z_c}{M_c} \\ \\ y_{ZMP} \end{bmatrix} \tag{4.5}$$

The ZMP equations in (4.4) are then discretized at a sampling period T, and expressed as a Linear Time Invariant (LTI) system as follows :

$$\begin{aligned} \mathcal{X}(k+1) &= A\mathcal{X}(k) + Bu(k) \\ p(k+1) &= C\mathcal{X}(k) \end{aligned} \tag{4.6}$$

where

$$\begin{aligned}
\mathcal{X}(k) &= [x_c(kT)\ \dot{x}_c(kT)\ \ddot{x}_c(kT)\ y_c(kT)\ \dot{y}_c(kT)\ \ddot{y}_c(kT)]^T \\
u(k) &= u(kT) = [\dddot{x}_c(kT) \quad \dddot{y}_c(kT)]^T \\
p(k) &= p(kT),
\end{aligned}$$

$$A = \begin{bmatrix} 1 & T & \frac{T^2}{2} & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & \frac{T^2}{2} \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} \frac{T^3}{6} & 0 \\ \frac{T^2}{2} & 0 \\ T & 0 \\ 0 & \frac{T^3}{6} \\ 0 & \frac{T^2}{2} \\ 0 & T \end{bmatrix} \tag{4.7}$$

$$C = \begin{bmatrix} 1 & 0 & -\frac{(m_o+M_c)\,Z_c}{M_c\,g} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & -\frac{Z_c}{g} \end{bmatrix}$$

We can then find the optimal feedback gain to minimize the following cost function :

$$J = \sum_{i=k}^{\infty} \{Q_e e(i)^2 + \Delta x^T(i) Q_x \Delta x(i) + R \Delta u^2(i)\} \tag{4.8}$$

where $e(i)$ is the error between the reference and the actual ZMP defined as follows :

$$\begin{aligned}
e(k) &= p^{ref}(kT) - p(kT) \\
&= \begin{bmatrix} x^{ref}_{ZMP}(kT) + \frac{\mu\, m_o\, Z_c}{M_c} \\ \\ y^{ref}_{ZMP}(kT) \end{bmatrix} - p(kT)
\end{aligned} \tag{4.9}$$

where x^{ref}_{ZMP} and y^{ref}_{ZMP} are the ZMP reference trajectories. Q_x, Q_e and R are, respectively, the weights applied on the state vector, on the ZMP error and on the input. The control law that minimizes the cost function J for N previewed steps is defined as :

$$u(k) = -G_i \sum_{i=0}^{k} e(k) - G_x\, x(k) - \sum_{j=1}^{N_L} G_p(jT)\, p^{ref}\left((k+j)\,T\right) \tag{4.10}$$

where $G_i \in \mathbb{R}^{2\times 2}$, $G_x \in \mathbb{R}^{2\times 6}$ and $G_p(jT) \in \mathbb{R}^{2\times 2}$ are the optimal gains.

This control law can take care of the two ZMP perturbation sources :

1. The first source is the constant shift caused by the cart friction, and is compensated by adding an offset of $\frac{\mu m_o Z_c}{M_c}$ to the reference trajectory of ZMP.
2. The second one is related to the transported mass inertia, and is compensated by applying the appropriate optimal gain matrices G_i, G_x and G_p.

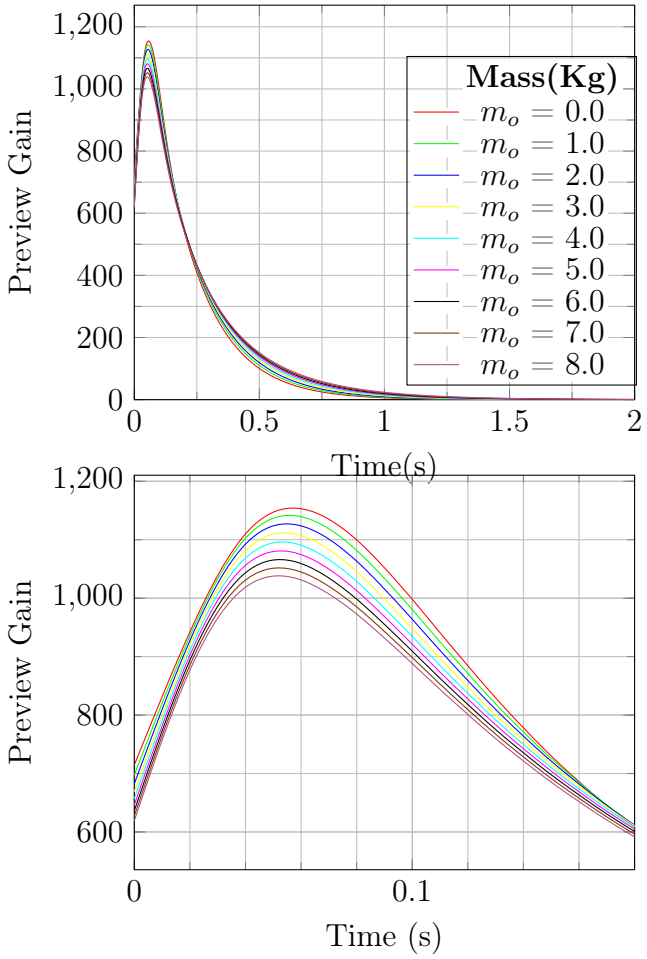


Figure 4.6 The norm of each matrix in the gains vector G_p for different mass and a close-up of the gain

4.4.2 Feedback Gain Analysis

The feedback gains (G_i, G_x and G_p) are usually computed offline since their values only depend on the matrices $\{A, B, C\}$ of the LTI system. However, in our case, the transported mass m_o appears in the C matrix, and thus those gains would need to be obtained each time a new mass is detected. Although it is possible to recompute the gains online, by solving Riccati equation, it adds a computational overhead that may be problematic since a gait controller must comply to strict real-time constraints.

As a first step, the preview gains were analyzed for a transported mass ranging between 0 and 8 kg, the latter being the maximum weight the cart can carry before compromising its structural integrity. Interval of 1 kg was chosen because our external force observer can only reliably detect a force within a 20% error margin and a 20% error on 4Kg, which is the mean of our usable weight range, is close to 1Kg. Fig. 4.6 presents the norm of the different matrices G_p in the preview gain according to the transported mass m_o. It is worth to point out that :

- The gains become close to zero at around $1.5s$ independently from the transported mass m_o.
- A difference of 1 kg does not seem to significantly change the gains.

Therefore, instead or recomputing the optimal gains, the controller will simply choose the more appropriate gain from this offline pre-computed set when a new mass is detected on the cart.

4.4.3 Controller Validation

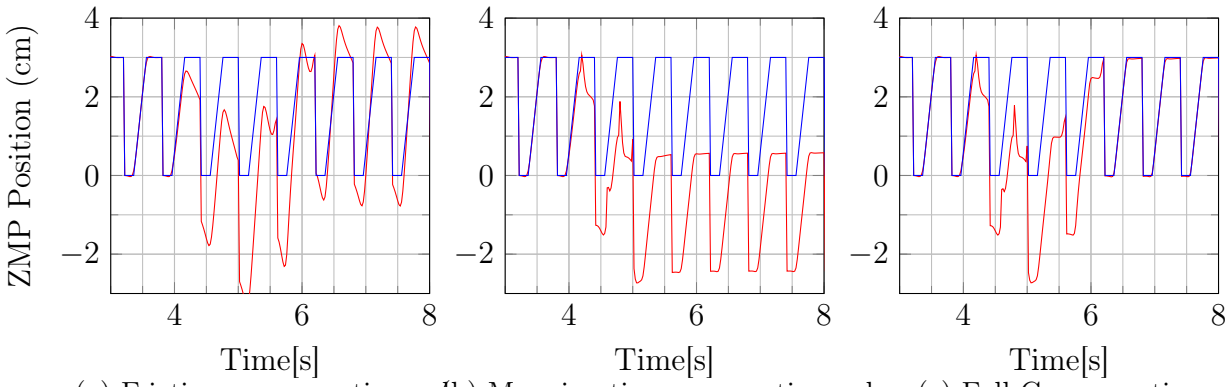
The controller was implemented and validated in Matlab using the model (4.4) presented in Section 4.3.4. Fig. 4.7 presents the local ZMP variation when a humanoid pushes a mass of 5Kg on a cart : (a) only the cart friction is compensated, (b) only the mass inertia is compensated and (c) full compensation of the external perturbation. As can be seen, the perturbation due to the mass inertia is resolved faster than that one due to the friction. This is because the optimal gains can be changed at anytime while the friction compensation is only done by modifying the reference ZMP at the end of the preview window, which is 1.5s in this case.

4.5 Experimental Result

Considering the good performance of the controller in simulation, it was implemented on Nao robot.

The controller validation is carried out using the following scenario :

- At the beginning, the robot walks in a straight path pushing an empty cart.
- A mass of 5kg is then added on the cart at $T = T_0$ while the robot continues to walk as usual.
- The robot estimates the mean of detected external forces over an interval of 4 steps (about 3 seconds) [Hawley et Suleiman, 2016].



(a) Friction compensation only (b) Mass inertia compensation only (c) Full Compensation

Figure 4.7 Simulation : position of the local ZMP in the sagittal plane of a humanoid pushing a cart. At approximately $T = 4s$, a mass of 5kg is added on the cart. The controller then modifies the walking gait to take into consideration the cart friction, the transported mass or both. As a result, the ZMP moves back to its natural position.

– At $T = T_0 + 5s$, the robot controller is triggered manually and the transported mass is estimated using Eq.(4.3), and then integrated into the walk pattern generator using the control law presented in Section 5.7.

Fig. 4.8 presents local ZMP measurements during the experiment. As can be seen, the controller is able to move the ZMP back towards its initial position ; thus can effectively be used to generate a more stable walking gait for a humanoid robot transporting a load with a rolling cart.

4.6 Conclusion

In this paper, we introduced a model-based controller for a humanoid pushing a heavy mass on a cart. As we demonstrated in simulation and experiments, the controller enables a humanoid to push a rolling cart more stably by effectively repositioning the ZMP towards the middle of the feet. Also, due to its low hardware (sensor) and computational requirements, the controller can be implemented on most humanoid robots.

Although path planning was not the focus of this work, the dynamic models developed in this paper could also be used in determining a feasible and optimal path while considering the limitations of the humanoid robot, the characteristics of the cart and the transported load.

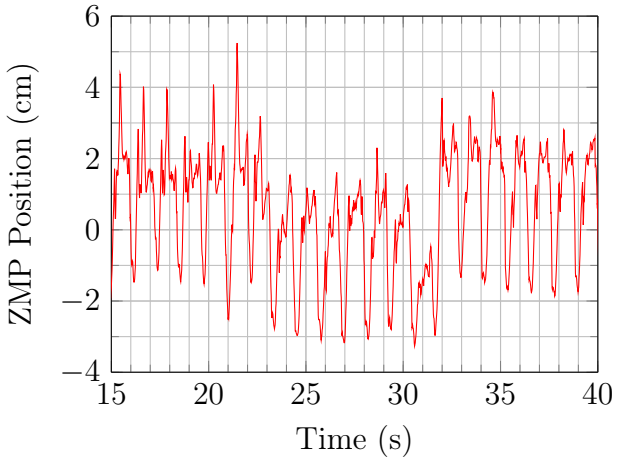


Figure 4.8 Local ZMP measurements during a cart-pushing experiment. At $T \approx 22s$, an unknown mass is added on the cart. At $T \approx 29s$, the controller modified the humanoid walking gait to take into consideration the cart friction and transported mass. As a result, the ZMP moves back to its natural position (regular walk) at $T \approx 32s$.

Acknowledgment

This research is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC).

CHAPITRE 5

Transport collaboratif par deux robots humanoïdes

Auteurs et affiliation :

L.Hawley : Étudiant à la maîtrise en génie électrique, Université de Sherbrooke, Faculté de génie, Département de génie électrique et informatique

W.Suleiman : Professeur, Laboratoire de robotique intelligente, interactive, intégrée et interdisciplinaire (IntRoLab), Université de Sherbrooke, Faculté de génie, Département de génie électrique et de génie informatique

État de la publication : En révision

Date de soumission : Mai 2018

Journal : Robotics and Autonomous Systems

Titre français : Stratégie de contrôle pour le transport collaboratif d'une charge par deux robots humanoïdes

Contribution au document : Cet article aborde directement la problématique principale étudiée dans le cadre de cette maitrise, le transport collaboratif d'une charge importante par deux robots humanoïdes. L'emphase est mise sur le contrôle de ce système par le biais d'une analyse de la dynamique ainsi que la conception d'un contrôleur optimal. Les algorithmes développés ont été validés avec deux robots humanoïdes de petite taille NAO transportant une variété d'objets jusqu'à un poids maximal de 2.23 Kg.

Résumé français : Dans cet article, une approche exhaustive destiné au transport collaboratif d'une charge importante par deux robots humanoïdes est présentée. Premièrement, un modèle dynamique simplifié de la tâche de transport collaboratif est développé et la stabilité du système est analysée rigoureusement. Ensuite, un contrôleur

centralisé est formulé, permettant le contrôle optimale du système en considérant la stabilité de chacun des robots ainsi que les contraintes induites par la tâche. Finalement, un planificateur local ainsi qu'un contrôle compliant des bras sont intégrés au contrôleur formant ainsi une solution complète permettant à deux humanoïdes de transporter une charge de façon stable. L'approche est analysée et validée en simulation en plus d'être mis à l'épreuve avec deux robots humanoïdes Nao transportant divers objets à l'aide d'une table. Les résultats expérimentaux valident notre approche en permettant notamment aux robots de transporter de façon stable certains objets imposants et lourd jusqu'à l'équivalent de la moitié du poids d'un robot.

5.1 Abstract

This paper aims at proposing a comprehensive control framework designed for cooperative transportation of a heavy load by two humanoid robots. First, a simplified dynamic model of the cooperative task is developed and the system stability is rigorously analyzed. Next, a centralized controller is formulated, this formulation provides an optimal control of the system by considering the robots dynamical stability while satisfying the robot-object-robot constraints. Finally, the controller is integrated with an arm controller and a local planner module forming a complete framework for cooperative transportation tasks. The approach is thoroughly analyzed and validated in simulation, and experiments are carried out on a team of two Nao humanoid robots transporting a range of objects placed on a small table. The experimental results pointed out the robustness of the approach as the robots successfully accomplished the transportation tasks in a stable way, moreover the transported objects masses were up to half the mass of one of the robot. Besides increasing the robot payload, some of the transported objects are relatively large and it is simply impossible for a single robot to transport them.

5.2 Introduction

The human-like structure of humanoid robots enables them to perform a variety of tasks, such as climbing a ladder and driving a car ; tasks that cannot be achieved using traditional wheeled robots. However, this increased mobility comes at a price as some of the simplest chores such as object manipulation and transportation can become challenging tasks. The main limitations with respect to object handling and transportation tasks come from humanoid robots delicate dynamic balance and the usually low power-to-weight ratio of the humanoid robot actuators .

Since object transportation is one of the most common tasks for a robot, it is important for humanoids to be able to efficiently handle them in order to enact their capacity in real-life situations.

To this end, one of the most interesting options is to have multiple robots cooperating in order to carry heavy and/or large object as shown in Fig. 5.1. Although simpler solutions such as using a wheeled-cart [Hawley et Suleiman, 2017; Rioux et Suleiman, 2018] can achieve great results, cooperative transportation has the advantage of not requiring any external tool, thus it is a promising option in precarious and unknown environments such as the scene of a natural disaster. However, cooperation between robots requires particular considerations at both motion planning and control levels.

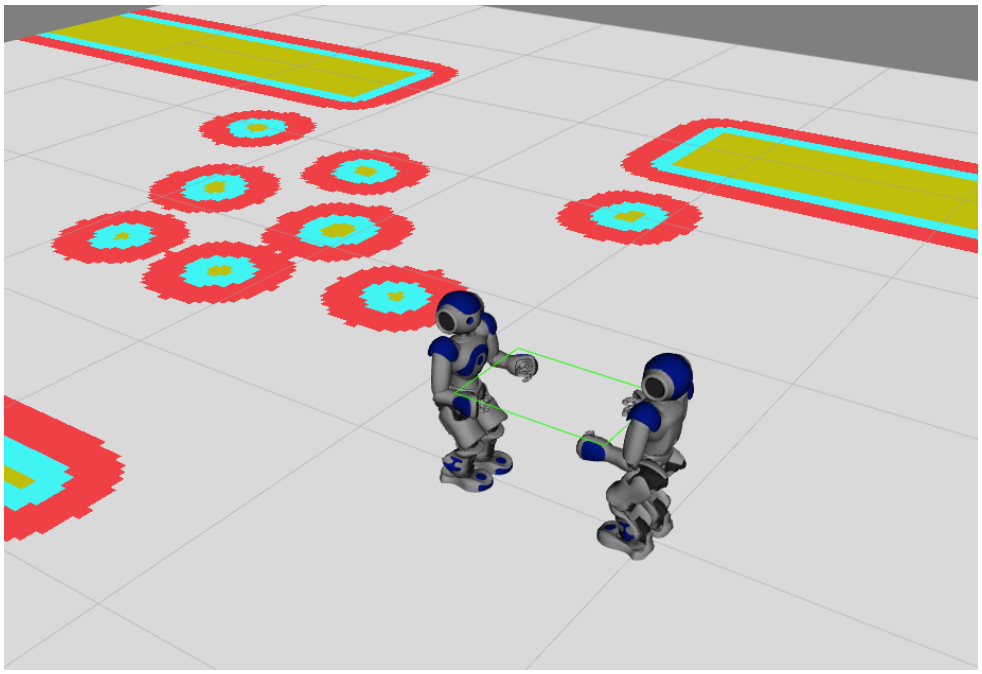


Figure 5.1 A team of two humanoid robots cooperating to transport an object among obstacles

5.3 Related Work

There are two main challenges that arise when two robots are rigidly linked together :

1. The robots motion planning modules must be linked and synchronized in such a way that they move in a coherent manner.
2. The robots controllers must also be connected to make sure they don't compete with each other while trying to maximize their own balance instead of ensuring the whole system stability and overall performance.

Previous works in both robot-robot and robot-human interaction can provide insights into how to deal with both challenges.

5.3.1 Leader-Slave Cooperation

There are multiple ways of planning the displacement of a robot-object-robot (or a human) system in the environment. The simplest one requires defining only the trajectory of one of the agents that is considered the leader. It is an approach that is often used in human-robot cooperation [Agravante *et coll.*, 2014; Bussy *et coll.*, 2012; Yokoyama *et coll.*, 2003] where the human acts as the leader and the robot simply follow along with the carried object acting as a haptic interface between the human and the robot. A drawback is that

complex coordinated motions that are usually required in a cluttered environment are not possible with this strategy. Furthermore, in a human-robot cooperation, the human can greatly reduce the imperfection in the haptic feedback by adapting his motions with the robot actions, however, in the case of a robot-robot scenario, the interaction force between the agents can increase rapidly. In [Wu *et coll.*, 2014], the Leader-Slave approach has been implemented on two HRP-2 robots with the slave robot using Force/Torque (F/T) sensors in the wrist to infer the intention/motion of the leader. Using this method, the leader robot velocity must be limited to consider the lag in the measurements of the force by the slave robot. Despite the relatively small velocity of the robots, strong interaction forces were measured during the experiments. Of course, it is expected that the results would be far worse if the robots were not equipped with F/T sensors as is the case for the majority of small or medium-sized humanoid robots.

5.3.2 Symmetry Control Framework

The authors in [Wu *et coll.*, 2016] proposed a control framework for multiple humanoids transporting an object where the input is a trajectory for the transported object and a desired internal force. Then, the trajectories for the hands of each robot are generated from the object desired trajectory and the robots footsteps are finally produced in order to keep the feet at a constant position from the center of the object.

From a motion planning point of view, it makes more sense to specify the trajectory of the object rather than the trajectory of one of the robots as done in the master-slave approach. However, keeping the robot always at a fixed position from each other severely reduce the set of feasible motions, some of which might be needed in complex environments.

An interesting aspect of the framework is the possibility of specifying an internal force as this force could be useful when transporting a deformable object. However, the authors only present the most practical case where the desired internal force is set to zero. The interaction force between a robot and the object is limited by moving the arms using a force control loop that uses measurements from F/T sensors in the robot wrists. As a consequence, the robustness of the framework highly depends on the performance of the force sensor akin to the master-slave approach. Furthermore, the pattern generator does not consider the whole system dynamic, but instead it is only treated as an inverse kinematics problem where constraints on the feet and the torso of a robot are computed by each robot pattern generator and the hand position constraints are supplied by a high-level motion planner. The approach exhibited a good performance in simulation with a

transported object of negligible mass and the interaction force between the robot kept low by using the force control loop of the arms.

In [McGill et Lee, 2011], the authors used two Darwin-OP robots to transport a small stretcher. The proposed approach used a separate control schemes for each robot where the footsteps were synchronized using communication over a wireless network. The distance between the robots was also monitored using the embedded camera of the robots. Therefore, the problem was treated more as a synchronization task in terms of footstep timing and relative displacement. However, the dynamic was simplified as the stretcher had a negligible mass, moreover, no interaction force was applied to the robots since the stretcher could easily slide into the robot grippers.

5.3.3 Centralized Controller

In [Rioux *et coll.*, 2015, 2017], a thorough strategy for planning is presented as the whole system footprint consisting of the two humanoid robots and of the transported object is considered in the motion planning phase. Using this approach, the robots can use their arms more effectively to execute more complex motions such as sharp turns in tight corners. The biggest challenge with this strategy is ensuring that the robots are accurately following their respective trajectory and in a synchronized manner. The strategy used by the authors is to separate the navigation into two tasks : the first one is for each robot to follow its respective trajectory and the second is to respect the desired distance between the robots. Therefore, each robot must locate itself in the world frame as well as with respect to the other robot. A similar approach was investigated in [Inoue *et coll.*, 2003], where all the possible positional errors (vertical misalignment, horizontal misalignment, etc.) between two humanoid robots are estimated and the optimal strategies to bring the system back to the desired state are found using Q-learning methods.

This sort of approach is efficient in cases where the agents are not directly linked together such as drone flying in formation, but as it has been shown by the authors of both papers, it is often not sufficient in the case of rigidly linked humanoid robots where the acceptable margin of error is really small.

Up to our knowledge, the most complete approach is probably the one presented in [Bouyarmane et Kheddar, 2012], where the authors propose a general framework for humanoid robots enabling them to execute any task by considering all the kinematic and dynamic constraints. Unfortunately, the computational requirements of this approach makes it currently unadapted for real-life scenarios.

5.4 Contributions

The main contributions in this work are :

- Designing a control framework allowing two humanoid robots to efficiently execute a cooperative transportation task of a heavy or large load. The stability of the closed-loop control system is also thoroughly analyzed.
- The implementation of the framework on two Nao humanoid robots. Even though the proposed framework is general enough to be applied to any humanoid robot, the presented implementation is mainly oriented towards small or medium-sized humanoid robots having limited hardware and computational capacities, such as a Nao robot.

The global architecture of the developed framework to carry out the cooperative transportation task is shown in Fig. 5.2. The next subsections briefly present each module separately.

5.4.1 Global Planner

The role of the global planner is to find a valid path for the system (robot-object-robot) to reach the desired goal (position and orientation). This can be a particularly complex task in the case of two rigidly linked humanoid robots since there are a lot of degrees of freedom. In our previous works [Rioux *et coll.*, 2015, 2017], we investigated the problem and came up with a lattice-based graph planner that uses motion primitives to reduce the set of possible motions. Using this approach, it is possible to easily tweak the planner to take into account the limitations and abilities of different robots. This paper does not focus on the global planner and the framework is independent of the actual implementation of this module as long as it provides a high-level command for the system whether it is an operator command or a global path provided by a planning algorithm.

5.4.2 Local Planner

The local planner module manages the navigation of the robots. In other words, the local planner provides a high-level command to the humanoid robots to follow the desired global plan as shown in Fig. 5.3. Typical high-level motion commands for humanoid robots are speed reference [Herdt *et coll.*, 2010] or footsteps [Garimort *et coll.*, 2011].

In the case of a cooperative transportation task, another role of the local planner is to ensure the synchronization of the robots displacements. Indeed, if one robot is progressing more rapidly than the other one for various reasons (slipping, external disturbance, etc.), the system will rapidly become unstable since the transported object ties

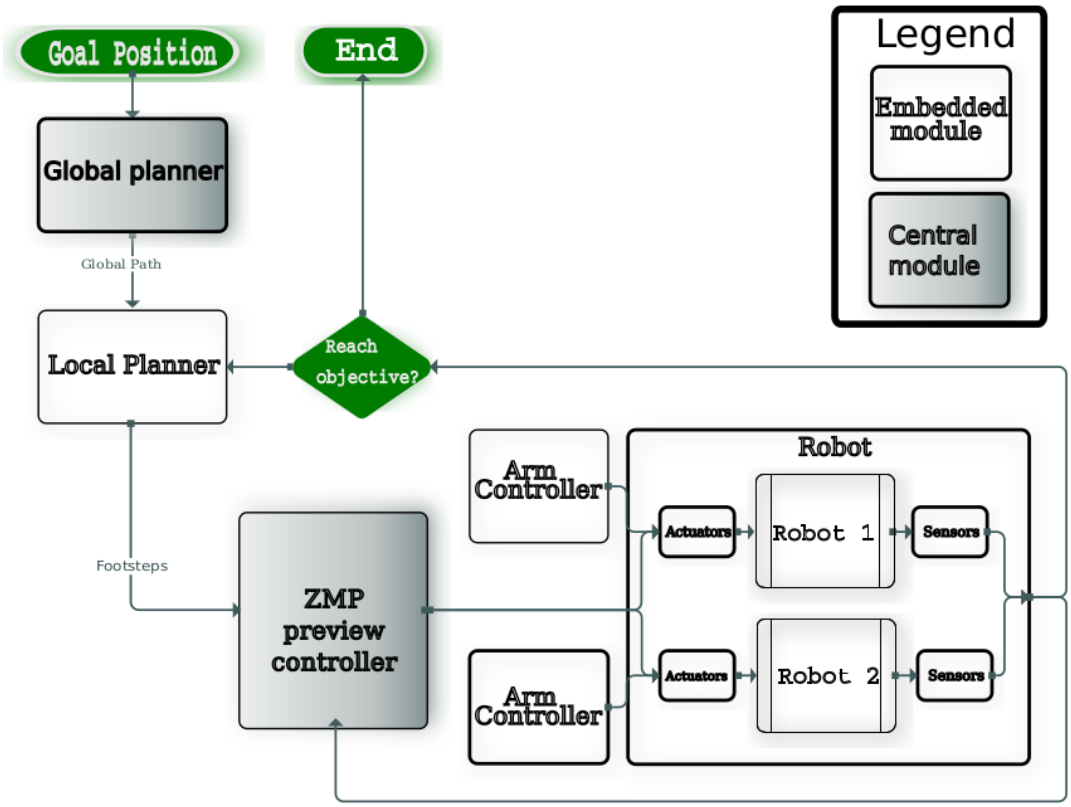


Figure 5.2 Global architecture : central modules are executed on an external PC and embedded modules are executed on the robot on-board computer.

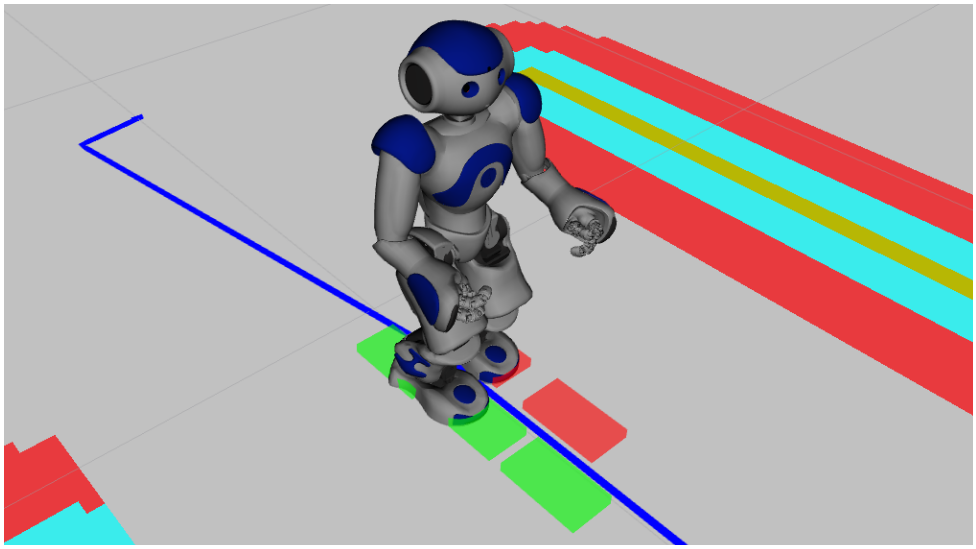


Figure 5.3 A footstep planner is used as a local planner to follow a global trajectory, which is in blue line, among obstacles

them together. In this case, changing the footsteps is the preferable high-level control command as it gives an accurate control over the displacement of each robot. The local planner module is discussed in section 5.8.

5.4.3 ZMP Preview Controller

The objective of the ZMP preview controller is to compute the joint trajectories for each robot in order to perform the high-level footstep command in a stable way. A popular approach is the model predictive control [Kajita *et coll.*, 2003] which uses a simplified model of the robot dynamics. This module is central since it has an immediate control over the robot motions contrarily to the local planner that can only modify the footsteps at discrete intervals. In our previous works [Rioux *et coll.*, 2015, 2017], we treated cooperative transportation as a kinematic problem since we didn't consider the particular dynamic of this task. Hence, the approach was validated with the robots carrying a flexible table of negligible weight, thereby limiting the interaction force. As the motion of each robot directly impacts the other one via the transported object, a centralized controller is the preferable option in this case. In Section 5.6, the dynamic of the cooperative transportation task is analyzed. Section 5.7 presents our controller formulation.

5.4.4 Arm Controller

In human-robot and robot-robot cooperative system, the arms often play an important role since the interaction between the agents usually occurs through them. It is also the case for cooperative transportation, and as such special attention is given to the control of the arms during the task. In Section 5.5, the particular dynamic behavior and stability of the humanoid-object-humanoid system are presented and the relevance of having compliant arms during the task is discussed.

5.5 Compliance Control of the Robot Arms

One of the biggest challenges with cooperative transportation by two humanoid robots is that commonly used stabilization strategies [Stephens, 2008] are no longer valid. This is because those approaches assume that all external forces applied to the humanoid robot are undesirable and must be compensated. However, when two robots are rigidly linked together, the main source of disturbance is the interaction between the robots. Therefore, it is no longer desirable to try to compensate for the force, but instead the robots must comply with it.

In such a situation, the controller must act rapidly to re-establish stability by reducing the perturbing interaction force. In a human-robot scenario, the human can quickly and actively rectify the situation and act accordingly. It is a more complex problem in a robot-robot collaborative transportation scenario. This is because the controller can either modify the desired ZMP trajectory or find a way to move the CoM. The first solution is done by modifying the footsteps to bring the robot back to the desired relative position with respect to its partner. The second option can be achieved by moving the arms, thereby maintaining control of the robot torso/CoM.

Modifying the footsteps is the only solution that can really bring the system back to its desired state, but it is quite slow. That is why an important part of any humanoid robot interaction system is the implementation of a compliance control. Indeed, most work [Don Joven Agravante, Andrea Cherubini, Alexander Sherikov, Pierre-Brice Wieber, 2017; Stasse *et coll.*, 2009; Wu *et coll.*, 2014, 2016] in robot-robot and human-robot cooperation have implemented some sort of compliance of the robot arms as a way to ensure the robot balance while executing the interaction task.

The most widely documented method in the literature for implementing arm compliance with position-controlled humanoid robot is based on an impedance control loop of the end-effector position. The control law is usually given by :

$$\begin{aligned} x_m &= x - x_r \\ F_x &= D\,\ddot{x}_m + B\,\dot{x}_m + K\,x_m \end{aligned} \tag{5.1}$$

where x_m is the difference between the actual, x, and desired position, x_r, of the hands, D is the desired (apparent) inertia, B is the desired damping coefficient and K is the desired stiffness coefficient.

Using this method and assuming that the external force F_x can be accurately measured, the arms of a position-controlled robot can become actively compliant. In most cases, the external force F_x is obtained with six-axis F/T sensors located in the robot wrists. Even though this approach has been proven to be efficient, a lot of humanoid robots are not equipped with such sensors, thus the external force must be estimated using less reliable methods [Berger *et coll.*, 2015; Hawley et Suleiman, 2016; Mattioli et Vendittelli, 2017]. In the next section, the dynamic of the cooperative transportation task by two humanoids is analyzed.

5.6 Dynamic Models

Let us consider a system consisting of two humanoid robots transporting a mass of m_0 as depicted in Fig. 5.4. In the sequel, we refer to the robot moving forward in the direction of the displacement as robot 1 and the robot moving backward as robot 2. It is assumed that the robots arms move under the influence of an external force modeled with the impedance control law in Eq.(5.1). Additionally, only the stiffness component in (5.1) is considered as it is assumed to be the main component of the compliance.

5.6.1 Dynamic Model of Two Humanoid Robots Transporting an Object

By approximating each robot by its LIPM model and modeling the impact of cooperative transportation task as an external force acting on the CoM of the LIPM model, the dynamic equation for robot 1 is given by :

$$M_{c_1}\,\ddot{x}_{c_1} = \frac{M_{c_1}\,g}{z_{c_1}}\,x_{c_1} + F_{x_1} \tag{5.2}$$

where :

- M_{c_1} and z_{c_1} are respectively the mass and the height of CoM.
- g is the magnitude of gravity acceleration.
- x_{c_1} and $\ddot{x}_{c_1}$ are respectively the position and the acceleration of the projection of robot 1 CoM on the x axis.

The position, velocity and acceleration, x_0, $\dot{x}_0$, $\ddot{x}_0$, of the projection of the carried mass on the x axis are given by :

$$\begin{aligned}
x_0 &= x_{c_1} + \frac{L}{2} + (x_{c_2} - x_{c_1} - L)\,\frac{K_{eq}}{K_1} \\
\dot{x}_0 &= \dot{x}_{c_1} + (\dot{x}_{c_2} - \dot{x}_{c_1})\,\frac{K_{eq}}{K_1} \\
\ddot{x}_0 &= \ddot{x}_{c_1} + (\ddot{x}_{c_2} - \ddot{x}_{c_1})\,\frac{K_{eq}}{K_1}
\end{aligned}$$

where

- x_{c_2}, $\dot{x}_{c_2}$, $\ddot{x}_{c_2}$ are respectively the position, velocity and acceleration of the projection of robot 2 CoM on the x axis
- L is the initial/desired distance between the two robots.

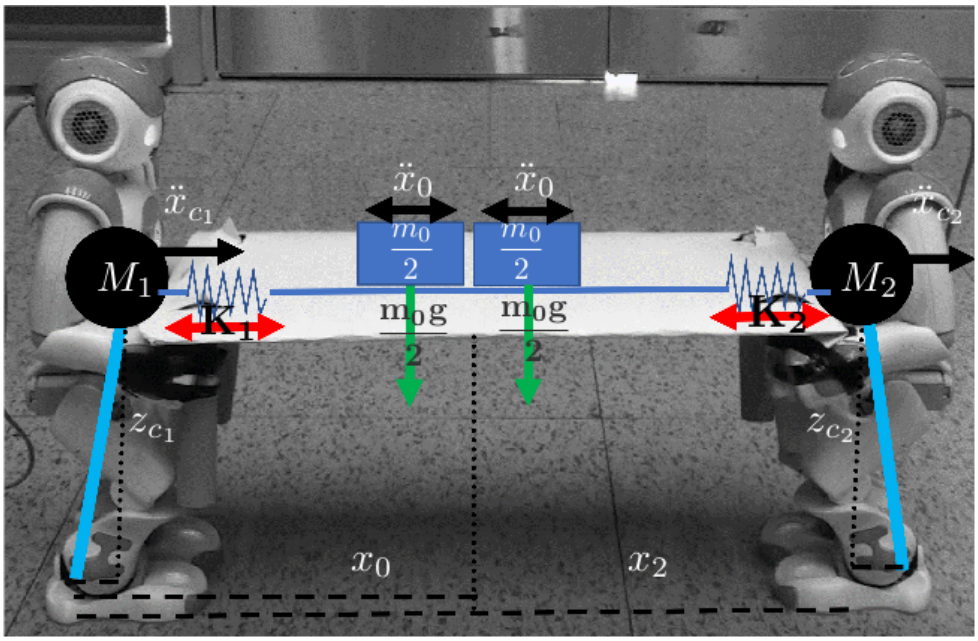


Figure 5.4 Dynamic model for two humanoid robots carrying a load

– K_{eq} is the equivalent stiffness coefficient :

$$K_{eq} = \frac{K_1 K_2}{K_1 + K_2}$$

and K_i is the stiffness coefficient of the robot i.

Without loss of generality, we assume that the carried mass m_0 is located at the middle of the transported object and is split uniformly between the two robots, a mass $\frac{m_0}{2}$ is therefore carried by each robot. Note that if the object is not placed in the middle, each robot should estimate its carried mass separately as explained in Section 5.9.2.

Then, the external force, F_{x_1}, applied to robot 1 along x axis can be expressed as :

$$F_{x_1} = -\frac{m_0}{2}\ddot{x}_0 + \frac{m_0\, g}{2\, z_{c_1}}\, x_1^h + K_{eq}\,(x_{c_2} - x_{c_1} - L) \tag{5.3}$$

where x_1^h is the position of the robot 1 hands :

$$x_1^h = x_{c_1} + x_1^{h^{ref}} + (x_{c_2} - x_{c_1} - L)\,\frac{K_{eq}}{K_1}$$

$x_1^{h^{ref}}$ is the initial/reference hands position with respect to the CoM.

In a similar way, the dynamic equation for robot 2 is given by

$$M_{c_2}\,\ddot{x}_{c_2} = \frac{M_{c_2}\;g}{z_{c_2}}\,x_{c_2} + F_{x_2} \tag{5.4}$$

where :

$$F_{x_2} = -\frac{m_0}{2}\ddot{x}_0 + \frac{m_0\;g}{2\,z_{c_1}}\,x_2^h \;-\; K_{eq}\,(x_{c_2} - x_{c_1} - L)$$

where x_2^h is the position of the robot 2 hands :

$$x_2^h = x_{c_2} + x_2^{h^{ref}} - (x_{c_2} - x_{c_1} - L)\,\frac{K_{eq}}{K_2}$$

(5.2) and (5.4) point out that there are two possible sources of disturbance in a cooperative transportation task. The first set of disturbances is caused by the transported load mass ($-\frac{m_0}{2}\ddot{x}_0 + \frac{m_0\;g}{2\,z_{c_{1,2}}}\,x_{1,2}^h$). The second one is caused by the interaction between the robots ($\pm K_{eq}\,(x_{c_2} - x_{c_1} - L)$). The next subsection investigates the impact of the abovementioned sources of disturbance on the system stability.

5.6.2 ZMP Dynamic for Two Humanoid Robots Transporting an Object

The CoP/ZMP is an important point for humanoid robot control, since it provides a good estimation of the robot stability. The ZMP dynamic equations for the cooperative transportation task can be found using the previously presented motion equations of each robot ((5.2) and (5.4)). First, let the ZMP along x axis of each robot be expressed by the following vector :

$$\boldsymbol{p} = \begin{bmatrix} p_{x_1} & p_{x_2} \end{bmatrix}^T$$

where p_{x_1} and p_{x_2} are respectively the projections of the ZMP of the robots 1 and 2 on the x axis.

As shown in [Agravante *et coll.*, 2016; Hawley et Suleiman, 2017], the ZMP is given by the following formula :

$$\boldsymbol{p} = \begin{bmatrix} p_{x_1} \\ \\ p_{x_2} \end{bmatrix} = \begin{bmatrix} x_{c_1} + \frac{M_{c_1}\,z_{c_1}}{f_{n_1}^o}\,\ddot{x}_{c_1} - \frac{z_{c_1}}{f_{n_1}^o}\,F_{x_1} \\ \\ x_{c_2} + \frac{M_{c_2}\,z_{c_2}}{f_{n_2}^o}\,\ddot{x}_{c_2} - \frac{z_{c_2}}{f_{n_2}^o}\,F_{x_2} \end{bmatrix} \tag{5.5}$$

where M_{c_i} is the total mass of the robot i, x_{c_i} and $\ddot{x}_{c_i}$ are the position and acceleration of the robot i CoM projection on the x axis. F_{x_i} is the external force applied to the CoM of robot i along x axis.

$f^o_{n_i}$ is the total normal force applied to the robot i feet, it is given by the following expression :

$$f^o_{n_i} = -M_{c_i}\, g - M_{c_i}\, \ddot{z}_{c_i} + F_{z_i} \tag{5.6}$$

where F_{z_i} is the external force applied to the CoM of robot i along z axis.

By assuming that the CoM height, z_{c_i}, is constant during the locomotion, and replacing $F_{z_i} = -\frac{m_0}{2}\, g$, we obtain :

$$\begin{aligned} f^o_{n_i} &= -\left(M_{c_i} + \frac{m_0}{2}\right) g \\ &\triangleq -\widehat{M_i}\, g \end{aligned} \tag{5.7}$$

As a result, p_{x_1} and p_{x_2} become :

$$\begin{aligned} p_{x_1} =& \frac{M_{c_1}}{\widehat{M_1}}\, x_{c_1} - \frac{M_{c_1}}{\widehat{M_1}\, g}\, \ddot{x}_{c_1} + \frac{m_0}{2\widehat{M_1}}\, x^h_1 - \frac{m_0\, z_{c_1}}{2\widehat{M_1}\ g}\, \ddot{x}_0 \\ &+ \frac{K_{eq}(x_{c_2} - x_{c_1} - L)}{\widehat{M_1}\ g} z_{c_1} \end{aligned} \tag{5.8}$$

and

$$\begin{aligned} p_{x_2} =& \frac{M_{c_2}}{\widehat{M_2}} x_{c_2} - \frac{M_{c_2}}{\widehat{M_2}\, g}\, \ddot{x}_{c_2} + \frac{m_0}{2\widehat{M_2}}\, x^h_2 - \frac{m_0\, z_{c_2}}{2\widehat{M_2}\ g}\, \ddot{x}_0 \\ &- \frac{K_{eq}(x_{c_2} - x_{c_1} - L)}{\widehat{M_2}\ g}\, z_{c_2} \end{aligned} \tag{5.9}$$

Similar equations can be found for the ZMP on the y axis.

5.7 Controller

Using the ZMP equations (5.8) and (5.9) of the robots, our objective is to define a state-space representation and formulate the control problem as an optimization problem.

5.7.1 State-Space Formulation

First, let us define a new variable x_2 such that :

$$\begin{bmatrix} x_2 \\ \dot{x}_2 \\ \ddot{x}_2 \end{bmatrix} = \begin{bmatrix} x_{c_2} - L \\ \dot{x}_{c_2} \\ \ddot{x}_{c_2} \end{bmatrix} \tag{5.10}$$

and p such that :

$$p(t) = \begin{bmatrix} p_{x_1} - \frac{m_0}{2\widehat{M_1}} x_1^{h^{ref}} \\ \\ p_{x_2} - L - \frac{m_0}{2\widehat{M_2}} x_2^{h^{ref}} \end{bmatrix} \tag{5.11}$$

Then, it is possible to discretize the ZMP equations in (5.8) and (5.9) at a sampling period T, and express the system as a Linear Time Invariant (LTI) system as follows :

$$\begin{aligned} \mathcal{X}(k+1) &= A\mathcal{X}(k) + Bu(k) \\ p(k+1) &= C\mathcal{X}(k) \end{aligned} \tag{5.12}$$

where :

$$\mathcal{X}(k) = [x_{c_1}(kT) \quad \dot{x}_{c_1}(kT) \quad \ddot{x}_{c_1}(kT) \quad x_2(kT) \quad \dot{x}_2(kT) \quad \ddot{x}_2(kT)]^T \tag{5.13}$$

$$\begin{aligned} u(k) &= u(kT) = [\dddot{x}_{c_1}(kT) \quad \dddot{x}_2(kT)]^T \\ p(k) &= p(kT), \\ A &= \begin{bmatrix} 1 & T & \frac{T^2}{2} & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & \frac{T^2}{2} \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} \frac{T^3}{6} & 0 \\ \frac{T^2}{2} & 0 \\ T & 0 \\ 0 & \frac{T^3}{6} \\ 0 & \frac{T^2}{2} \\ 0 & T \end{bmatrix} \end{aligned} \tag{5.14}$$

$$C = \begin{bmatrix} \frac{M_{c_1}+(1-\frac{K_{eq}}{K_1})\frac{m_0}{2}}{\widehat{M_1}} - \frac{z_{c_1} K_{eq}}{\widehat{M_1}\, g} & \frac{z_{c_2} K_{eq}}{\widehat{M_2}\, g} + \frac{\frac{K_{eq}}{K_2} m_0}{2\,\widehat{M_2}} \\ 0 & 0 \\ -\frac{M_{c_1}\, z_{c_1}}{\widehat{M_1}\, g} - \frac{[1-\frac{K_{eq}}{K_1}]m_0\, z_{c_1}}{2\,\widehat{M_1}} & -\frac{(1-\frac{K_{eq}}{K_1})m_0\, z_{c_2}}{2\,\widehat{M_2}} \\ \frac{z_{c_1} K_{eq}}{\widehat{M_1}\, g} + \frac{\frac{K_{eq}}{K_1} m_0}{2\,\widehat{M_1}} & \frac{M_{c_2}+(1-\frac{K_{eq}}{K_2})\frac{m_0}{2}}{\widehat{M_2}} - \frac{z_{c_2} K_{eq}}{\widehat{M_2}\, g} \\ 0 & 0 \\ -\frac{\frac{K_{eq}}{K_1} m_0\, z_{c_1}}{2\,\widehat{M_1}} & \frac{-M_{c_2}\, z_{c_2}}{\widehat{M_2}\, g} - \frac{\frac{K_{eq}}{K_1} m_0\, z_{c_2}}{2\,\widehat{M_2}} \end{bmatrix}^T \tag{5.15}$$

The main objective of the controller is to maximize the system stability by maintaining the ZMP of each robot at the center of the supporting foot. This can be expressed as an optimization problem where the cost function to be minimized is expressed as :

$$J = e^T\, Q_e\, e + \mathcal{X}^T\, Q_x\, \mathcal{X} + u^T\, R\, u \tag{5.16}$$

where e is the error between the reference and the actual ZMP defined as follows :

$$\begin{aligned} e &= p^{ref} - p \\ &= \begin{bmatrix} p_{x_1}^{ref} - \frac{m_0}{2\,\widehat{M_1}}\, x_1^{h\,ref} \\ \\ p_{x_2}^{ref} - L - \frac{m_0}{2\,\widehat{M_2}}\, x_2^{h\,ref} \end{bmatrix} - \begin{bmatrix} p_{x_1} - \frac{m_0}{2\,\widehat{M_1}}\, x_1^{h\,ref} \\ \\ p_{x_2} - L - \frac{m_0}{2\,\widehat{M_2}}\, x_2^{h\,ref} \end{bmatrix} \\ &= \begin{bmatrix} p_{x_1}^{ref} - p_{x_1} \\ \\ p_{x_2}^{ref} - p_{x_2} \end{bmatrix} \end{aligned} \tag{5.17}$$

$p_{x_1}^{ref}$ and $p_{x_2}^{ref}$ are the ZMP reference trajectories of each robot.

The matrices Q_x, Q_e and R are positive definite. This cost function means that we wish to follow the reference ZMP trajectory without rapid change in the states and the inputs.

5.7.2 Preview Control

Model predictive control approach is one of the most used methods in gait generation for humanoid robots. It uses a model of the robot dynamic to predict the effect of current command on the robot future states. The objective of the controller is then to minimize the cost function over a preview horizon N instead of only the current state such as :

$$\begin{aligned} \underset{u}{\text{minimize}} \quad & J = \sum_{i=k}^{N+k} \{e(i)^T\, Q_e\, e(i) + \mathcal{X}(i)^T\, Q_x\, \mathcal{X}(i) + u(i)^T\, R\, u(i)\} \\ \text{subject to} \quad & \mathcal{X}(k+1) = A\mathcal{X}(k) + Bu(k) \\ & p(k+1) = C\mathcal{X}(k) \end{aligned} \tag{5.18}$$

where $e(i)$ is the error between the reference and the actual ZMP defined as follows :

$$\begin{aligned} e(k) &= p^{ref}(kT) - p(kT) \\ &= \begin{bmatrix} p^{ref}_{x_1}(kT) - p_{x_1}(kT) \\ \\ p^{ref}_{x_2}(kT) - p_{x_2}(kT) \end{bmatrix} \end{aligned} \tag{5.19}$$

The next subsections present how to design a controller that minimize such a cost function.

Linear Quadratic Regulator

One of the most used approaches to solve (5.18) is Linear Quadratic Regulator (LQR) algorithm, which provides the optimal state-feedback, i.e. $u(k)$. Even though the formulation of (5.18) is under the finite-horizon form, many approaches, such as [Kajita *et coll.*, 2003], solve it as an infinite-horizon problem by assuming that N is big enough. This is because the Discrete time Algebraic Riccati Equation (DARE), in the case of infinite-horizon, is solved only once offline and used online for computing the optimal feedback, therefore it is particularly interesting for small to medium-sized humanoids with limited on-board computational power.

Preview Control as QP Problem

Another possible way to design the controller is to reformulate the optimization problem (5.18) as a Quadratic Programming (QP) problem of the following form :

$$\begin{aligned} \underset{u}{\text{minimize}} \quad & \frac{1}{2}u^T H u + g^T u \\ \text{subject to} \quad & b^- \leq Au \leq b^+ \\ & u^- \leq u \leq u^+ \end{aligned} \tag{5.20}$$

where H is a $n \times n$ is positive semidefinite matrix. The objective is to find the input vector u that minimizes the cost function while also satisfying some constraints.

The authors in [Herdt *et coll.*, 2010] present the detailed procedure to setup the QP problem in order to solve the cost function of (5.18). Evidently, the main advantage of using quadratic programming to solve this problem is the possibility to add constraints, such as constraining the ZMP within a certain area instead of following a predefined trajectory. However, this approach is computationally more expensive than the LQR algorithm even though it can usually be solved rapidly with an off-the-shelf solver such as [Ferreau *et coll.*, 2014]. Note that when no constraints are added, the QP can be solved analytically as shown in [Wieber, 2006].

5.7.3 Controller Robustness

The primary objective of the previously presented controller formulation is to ensure the stability of the robots by ensuring that the ZMP is kept at the center of the feet. There is however another aspect that must be considered, the robots CoM motions are actually restricted since the arms motions are limited to a small maneuverability range. Once the limits are reached, the compliance is lost and the system of robot-object-robot becomes rigidly linked. In this state, the system can easily collapse as explained in Section 5.5.

To avoid this situation, there are two ways of limiting the distance between the robots :

- Option 1 : Reducing the interaction force by implementing a fast and efficient compliance control of the arms.
- Option 2 : Relaxing the constraint of precisely following the ZMP reference trajectory by reducing the norm of weighting matrix on the error, $e(k)$, in (5.18).

The best option is of course option 1, however, implementing such control is a challenging task for small humanoid robots since it requires a rapid and accurate estimation of the interaction force and a fast motion of the actuators.

To increase the overall robustness of the system to positional errors, we propose to integrate an additional constraint on the relative torso displacement, hence allowing the ZMP to slightly move away from the reference position in order to reduce the torso displacement.

To this end, we add a new objective in the cost function, thereby obtaining :

$$J = e^T Q_e \, e + \mathcal{X}^T Q_x \, \mathcal{X} + u^T R \, u + (x_{c_1} - x_2)^T Q_d \, (x_{c_1} - x_2) \tag{5.21}$$

where Q_d is the weighting matrix on the new objective that minimizes the relative torso displacement between the two robots. With this simple formulation, the limitations on the arm motions can be implicitly integrated into the controller. Additionally, the limitation can be explicitly considered in the QP problem by adding the following constraint :

$$e^- \leq x_{c_1} - x_2 \leq e^+ \tag{5.22}$$

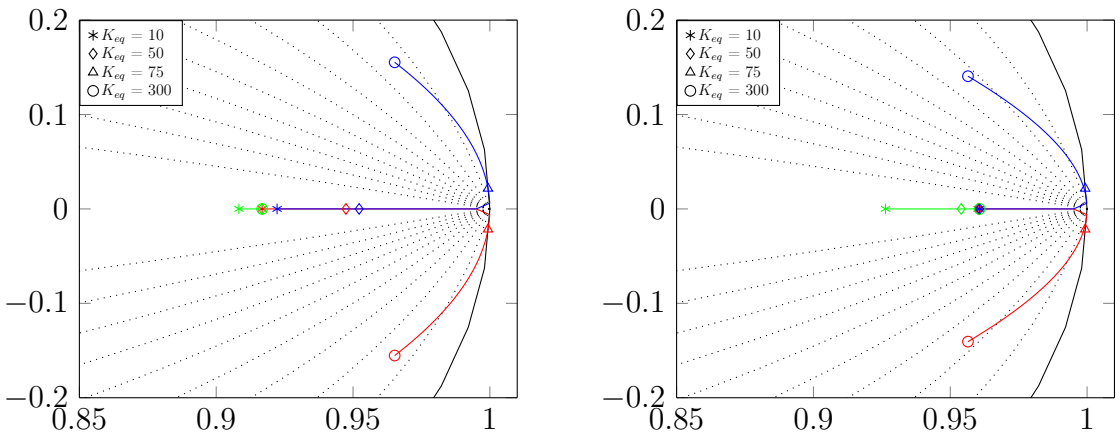
where e^- and e^+ are respectively the minimum and maximum relative displacement between the robots torsos.

5.7.4 Controller Stability

Another aspect to be taken into consideration during the design of the controller is the system stability and how it is affected by the transported load m_0 and the robots interaction (K_{eq}). Fig. 5.5 shows the closed-loop system poles for a K_{eq} in the range of 10 to 300 Nm^{-1} and two different transported masses.

The following points need to be kept in mind when designing the controller :

- The system is always stable. However, some values of K_{eq} are undesirable since they bring the poles really close to the instability region. In the case of small humanoid robots ($M_1 = M_2 = 5.1Kg$), the critical values are roughly between 70 and 80 Nm^{-1}.
- Fig. 5.5 points out that the oscillations in the closed-loop response can be avoided by keeping the interaction coefficient (K_{eq}) low.
- Transporting a heavier load has a stabilizing effect on the system response since it reduces the oscillations at high values of K_{eq}.



(a) $m_0 = 0Kg$ (b) $m_0 = 5Kg$

Figure 5.5 Closed-loop dominant pole positions for K_{eq} varying from 10 to 300.

5.8 Local Planner for Cooperative Transportation

Although the previously presented architecture of ZMP preview controller can ensure the robots stability, it cannot bring the system back to the desired state contrarily to some other control schemes that enable reactive footstep modification [Choi *et coll.*, 2015; Griffin *et coll.*, 2017; Stasse *et coll.*, 2009]. Therefore, it is the role of the local planner to ensure that both robots follow their respective paths in a stable way.

In order to accomplish these two objectives, the local planner is decomposed into two modules as presented in Fig. 5.6.

The first module is a conventional footstep planner [Garimort *et coll.*, 2011; Hornung *et coll.*, 2012] that, using the robots current position estimations and the global path, outputs high-level footsteps command to follow the global path. Since it can be difficult for a robot to position itself precisely in the world, another module is necessary to ensure that the robots do not move too close or too far from each other. The approach used here is to modify the footstep length, but another solution could have been to modify the step frequency of the robots [Motahar *et coll.*, 2015]. The used module is a finite-state machine (FSM) that modifies the footsteps command according to the robots relative position error.

As can be seen in Fig. 5.7(a), there are five possible states :

State 1. The robots Relative Position error (RP_{err}) is small. No compensation is needed.

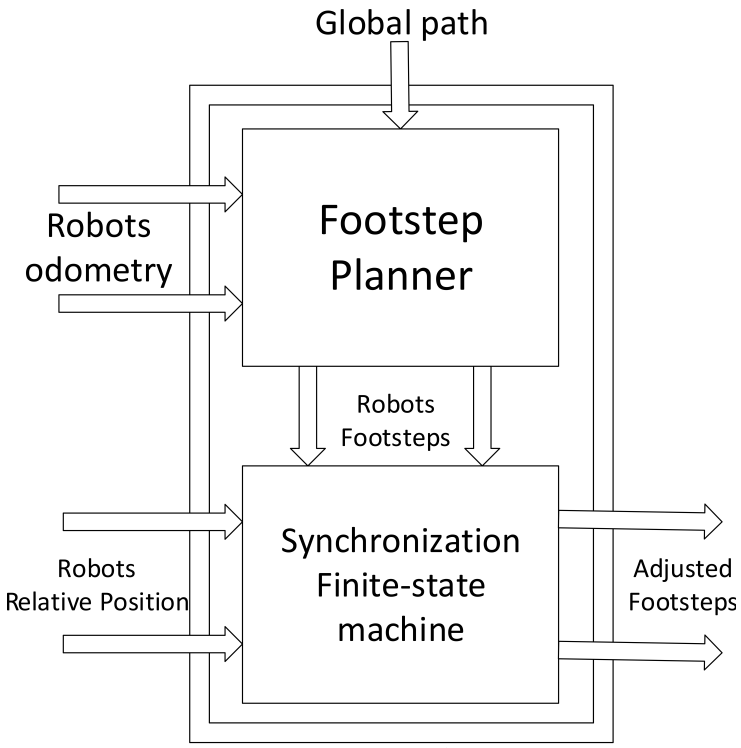


Figure 5.6 Block diagram of local planner

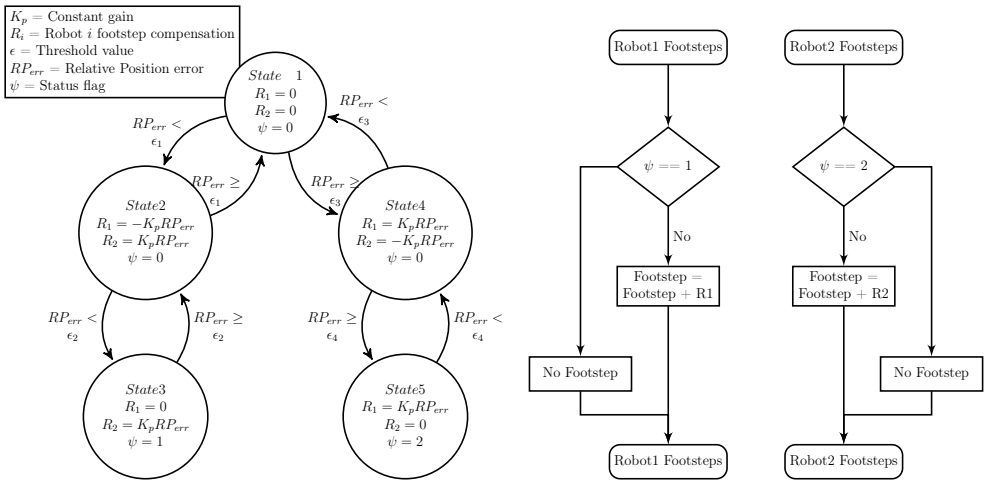
State 2. The robots are too close to each other ($RP_{err} < \epsilon_1$). The leading robot (robot 2) needs to speed up by taking bigger steps while the other robot will slow down by taking smaller steps.

State 3. The stability of the system is compromised as the arms are close to the limit ($RP_{err} < \epsilon_2$). The leading robot needs to speed up by taking bigger steps while the other robot should stop walking and wait for the system to come back to a more stable state.

State 4. The robots are too far from each other ($RP_{err} > \epsilon_3$). The leading robot needs to slow down by taking smaller steps while the other robot will speed up.

State 5. The stability of the system is compromised as the arms are extended to the maximum operational space ($RP_{err} > \epsilon_4$). The leading robot will stop walking while the other robot will walk faster until the system comes back to a more stable state.

The thresholds ϵ_1 and ϵ_3 are added so that compensation is done only when it is really needed. Whenever the footsteps are modified, the new resulting footprints must be checked for collision using a map of the environment. As for the thresholds ϵ_4 and ϵ_5, they were added because of the important slipping of the feet of Nao robot that was observed during the experimentations.



(a) Robots Synchronization FSM. (b) Footstep modification procedure

Figure 5.7 Block diagrams of footstep modification procedure in the synchronization module.

As shown in Fig. 5.6, the local planner needs both the estimated position of the robots in the world frame, using the robot odometry (visual or other), as well as an estimation of the robots relative position. An important assumption is that the robots are able to estimate their relative position more accurately than their global position.

In order to estimate the position between the two robots, two techniques were considered :

- Vision-based technique
- Closed-loop kinematic

In the previous version of our framework [Rioux *et coll.*, 2015], the global position of the robots was monitored using a Simultaneous Localization and Mapping (SLAM) algorithm. The robots relative position was then computed and compared with the already known object dimension to compute the relative position error. The precision of the estimation was therefore highly dependent on the performance of the localization algorithm, which turns out not to be enough for our application.

The approach was refined by adding a locally running optical flow based module [Rioux *et coll.*, 2017] that monitored the relative displacement of both robots using the robot embedded camera. A relatively good performance was obtained using this method but there were some drawbacks. First, the approach required some pre-processing to find a set of good features to track and to set the reference of the robots relative position. Second, an obvious requirement is that the robot can see each other. This requirement goes against the objective of cooperative transportation of a heavy or large object.

Using the knowledge of the object dimension, one can compute the transformation between the robot torso i and the center of the transported object as follows :

$$T_{ob}^{r_i} = midpoint(T_{r_i}^{hr} \times T_{hr}^{ob}, T_{r_i}^{hl} \times T_{hl}^{ob}) \tag{5.23}$$

where $T_{ob}^{r_i}$ is the homogeneous transformation matrix between the torso of robot i and the center of the object, $T_{r_i}^{h(l/r)}$ is the transformation between the robot torso and its left/right hand and $T_{h(l/r)}^{ob}$ is the transformation between the robot hand and the center of the object. However, this time we assume that $T_{h(l/r)}^{ob}$ is the actual transformation matrix and not simply a desired one. This assumption is valid because the table is rigid. Therefore, the relative position between the robots can be easily obtained with :

$$T_{r_2}^{r_1} = T_{ob}^{r_1} \times (T_{ob}^{r_2})^{-1} \tag{5.24}$$

If the object dimensions are precisely known, the performance of the method is only limited by the precision of kinematic model of the robot arms and the joint encoders. In the case of Nao robot, the available magnetic rotary encoders have a 12 bit precision that provides precision around $0.1°$. Considering the small size of Nao robot, a maximum error within a couple of mm is to be expected for the arms forward kinematics. Another advantage is that the method does not require a line of sight between the robots, thus allowing the robots to transport a huge object while allocating the cameras to other tasks, navigation for instance.

In the local control loop, it is not required to have an absolute transformation between the robots, but instead each robot only monitors its arms positions with respect to a reference/desirable position such as :

$$T_{r_i}^{h} = midpoint(T_{r_i}^{hr}, T_{r_i}^{hl}) \tag{5.25}$$

Then, the rotation and position errors between the actual hand position and the reference position for each robot can be found :

$$\begin{aligned} R_e &= R^{h^{ref}} \times R^{h^T} \\ P_e &= P^{h^{ref}} - P^h \end{aligned} \tag{5.26}$$

where R_e is a rotation matrix that expresses the orientation error, $R^{h^{ref}}$ is the desired orientation and R^h is obtained from the current transformation matrix $T_{r_i}^h$. P_e, $P^{h^{ref}}$

and P^h are respectively the positional error, reference position and actual position of the robot hands. All the rotation and position are expressed in the robot reference frame. These errors can then be used by the controller and by the local planner to adjust the footsteps and bring the system back to the desired state.

5.9 Implementation

The framework was tested and validated on a team of two Nao humanoid robots transporting an object placed on a light and rigid table. The next subsections describe the details of the implementation on the Nao robot.

5.9.1 Compliance Control of Nao Robot Arms

As mentioned in Section 5.5, the arms compliance is a critical component of robot-robot cooperation. In that section, the most common active compliance implementation method relies on F/T sensors measurements was presented.

However, regarding the Nao robot, a major implementation challenge is the estimation of the external force applied to the robot using the available sensors. Although good results were obtained using the method presented in [Hawley et Suleiman, 2018], the performance degrades during the cooperative transportation task. The main problem is that the force observer relies on the IMU, motor encoders and Force Sensitive Resistors (FSR) under the robot feet. However, during a cooperative transportation task, the interaction between the robots often causes a bad or lost contact of the feet on the floor which results in unreliable data from the FSR.

Thereby, to implement a compliance control of Nao arms, the force observer [Hawley et Suleiman, 2018] was used in combination with a joint level compliance strategy. It is possible to estimate the magnitude and direction of an external force applied to a manipulator if an accurate dynamic model of the robot is available and if the torques applied by the motors are known [Magrini *et coll.*, 2014, 2015]. The approach can also be extended to humanoid robots [Mattioli et Vendittelli, 2017], albeit it is more difficult since this type of robot is not rigidly attached to a base.

First, the torque due to the robot mass and transported object must be isolated. To do so, it is possible to use the following inverse dynamic equation :

$$\tau = B(q)\,\ddot{q} + C(q,\dot{q})\,\dot{q} + g(q) + J^T(q)\,F \tag{5.27}$$

where J is the Jacobian matrix of the considered chain, τ is the joint torques, $B(q)$ is the inertia matrix, $C(q, \dot{q})\,\dot{q}$ includes the Coriolis and centrifugal forces, $g(q)$ is the gravity terms, F is the external force applied at the end-effector, and $q, \dot{q}, \ddot{q}$ are respectively the position, velocity and acceleration of the joints.

The equation can be simplified since we estimate the torque when the arms are not moving ($\dot{q} = \ddot{q} = 0$) and we do not consider external forces. Therefore, we obtain :

$$\hat{\tau} = g(q) \tag{5.28}$$

The external force can then be easily obtained by subtracting this torque from the measured one to get the external force applied to the humanoid arms :

$$\tau_m - \hat{\tau} = J^T(q)\,F \tag{5.29}$$

where τ_m is the measured torques.

In [Mattioli et Vendittelli, 2017], the authors applied the above approach to a Nao robot to estimate a force applied anywhere on the body of the robot. Although great results were obtained in simulation, the implementation on a real Nao robot was rather inaccurate mostly because of the large static friction at the joints of the robot. However, in our case, we can greatly simplify the problem since we know the point of application of the interaction force. In a natural body configuration, in which the upper and lower arms are perpendicular as depicted in Fig. 5.1, an external force applied to the hands in the x-axis can be observed using only a single motor in the arm.

We decided to implement the arms compliance as an error minimization problem at the joint level. Referring to Nao robots official documentation [Aldebaran, 2016], we considered the shoulder pitch motor position error as an estimation of the interaction force. This is because the Nao arm motors are controlled through a PD controller, making it possible to determine the motor output torque with the following linear approximation [Mattioli et Vendittelli, 2017] :

$$\tau_m = K\,(q_d - q) \tag{5.30}$$

where $(q_d - q)$ is the position error and K is a constant coefficient that depends on multiple parameters of the joint motor (e.g. reduction ratio). Compared to our observer [Hawley et Suleiman, 2018], the motor error signal $(q_d - q)$ is less noisy, and more importantly only detects the forces that are applied to the robot hands. However, as pointed out in [Mattioli et Vendittelli, 2017], the error signal is only meaningful with a large interaction

force that produces a relatively high signal to noise ratio. Therefore, this strategy was used in conjunction with our force observer as depicted in Fig. 5.8.

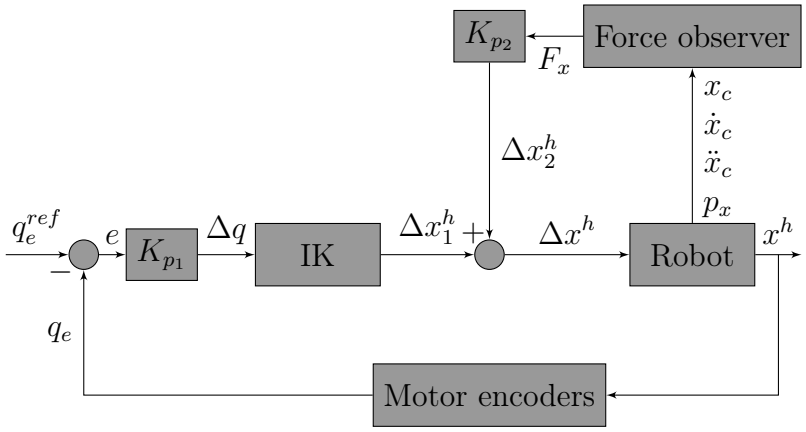


Figure 5.8 Arm Compliance Controller

Fig. 5.8 shows the control loop used to implement a compliant behavior of the Nao robot arms. The main component is a feedback loop that keeps the shoulder pitch motor position error (q_e) close to a reference value (q_e^{ref}) by moving the arms/hands (x^h) in the direction of the errors. K_{p_1} and K_{p_2} are proportional gains and IK stands for Inverse Kinematics algorithm that converts the shoulder displacement (q) into an equivalent hand displacement (Δx^h), the aim being to move the shoulders while keeping the hands in a parallel plane with respect to the floor. The secondary component is a feedforward compensation that is computed using the output from our external force observer [Hawley et Suleiman, 2018], F_x. The estimated force F_x is converted into a relative hands displacement (Δx_2^h) using a proportional gain and injected into the main loop. This architecture increases the controller robustness as the arms will only move rapidly if a force is detected in both the joint and the operational spaces. However, because of the modeling errors and in order to have a stable compliance controller, we added threshold values to both e and F_x, that means if e and F_x are within a set of thresholds, their values are not used and simply considered as 0.

The performance of the compliance controller was validated by a spring-based force sensor as well as by our external force observer [Hawley et Suleiman, 2018]. As shown in Fig. 5.9, a stiffness coefficient of 50 Nm^{-1} seems to model the compliance mechanism reasonably well, therefore we used that value in our implementation.

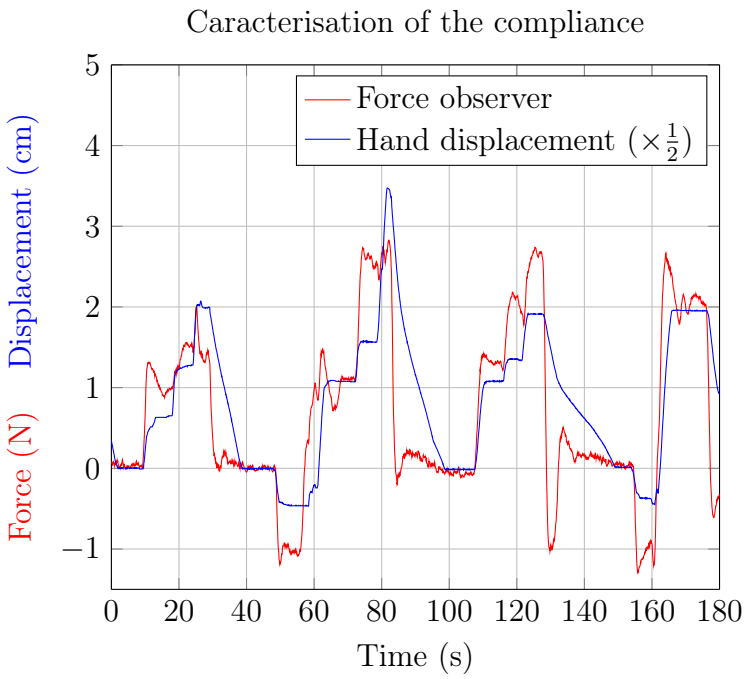


Figure 5.9 Displacement of the arms and the estimated external force using [Hawley et Suleiman, 2018]

5.9.2 Estimating the mass

The limited sensor set of the Nao robot has motivated the development of a model-based controller instead of solely relying on sensor measurements to estimate the system state vector. However, similarly to the robots relative distance estimation during the execution of the task, the transported mass is also an important part of the model that must also be estimated.

To this end, we used the external force observer described in [Hawley et Suleiman, 2018]. The transported mass could then be directly estimated using the observed force along the vertical axis, F_z, as follows :

$$m_0 = \frac{F_z}{g} \tag{5.31}$$

Fig. 5.10 presents the force observer outputs of one of the robots when they are immobile and a mass of 1.13 kg is added in the center of the transported rigid table. As can be seen, F_z is estimated at around $6\,N$ whereas the expected value is $5.54\,N$ by assuming that the mass is uniformly split between the two robots.

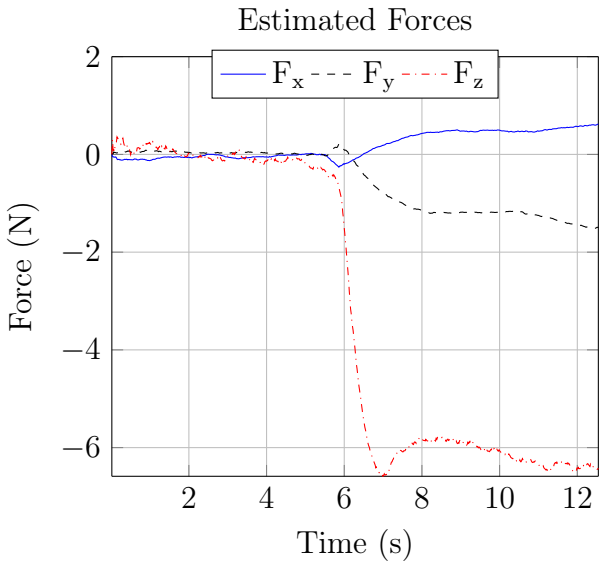


Figure 5.10 Estimated force by the force observer [Hawley et Suleiman, 2018] when a mass of 1.13 kg is placed on the transported table.

Once the mass is found, the optimal feedback gains (or the matrices used in the QP formulation) are computed once and can then be used for the whole task. It is also possible to precompute a set of gains for different masses as shown in [Hawley et Suleiman, 2017].

5.10 Experiments and Results

Before testing the system on real robots, the soundness of the controller was first asserted in simulation.

5.10.1 Controller Validation (Simulation)

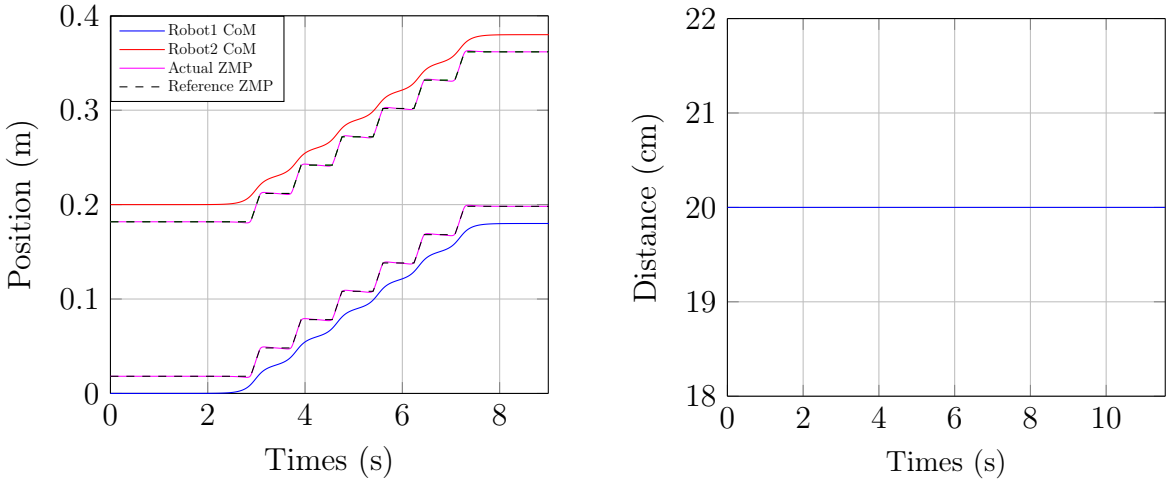
We have conducted two simulations :

- Case 1 : The robots need to get closer or farther away from each other.
- Case 2 : An external disturbance generates a rapid positional error between the robots.

The first case may be observed in a situation where the robot footprints are highly restricted, for instance in a cluttered environment. In this situation, the robots must remain stable despite the generated internal force. The second situation simply represents a case where one of the robot slips and an internal force is rapidly introduced in the system. The goal is then to bring the system back to the desired state while remaining stable.

The following parameters, which represent realistic values for our Nao robots, are used in simulation : $g = 9.81\,m.s^{-2}$, $T = 16.7\,ms$, $M_{c_{1,2}} = 5.1\,kg$, $K_{1,2} = 100\,Nm^{-1}$, $m_0 = 1\,kg$, $z_{c_{1,2}} = 0.315\,m$, $L = 0.20\,m$, $x_1^{ref} = 0.1\,m$, $x_2^{ref} = -0.1\,m$.

As for the controller, we used the QP approach with the following parameters : $N = 1.2\,s$, $Q_e = 1$, $R = 10^{-6}$, $Q_x = 0$ and $Q_d = 0$.



(a) Global ZMP and CoM trajectories in the sagittal plane (x-axis)

(b) The distance between the robots CoMs during the displacement

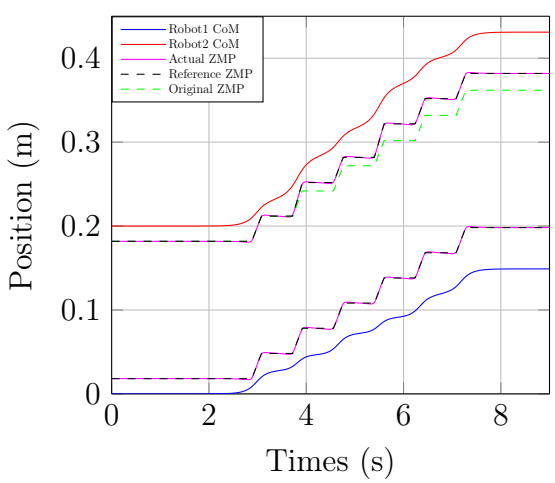
Figure 5.11 Simulation result for the optimal scenario where the robots are moving in a symmetrical way.

In all scenarios, the reference ZMP trajectories are generated so as to keep the ZMP in the middle of the support foot of each robot. Fig. 5.11 presents the ideal case where both robots are given symmetrical footstep commands at the same time. As can be seen, both robots are able to follow their respective reference ZMP trajectory and move in a perfectly symmetrical motion with their torsos remaining at the same relative distance during the whole trajectory.

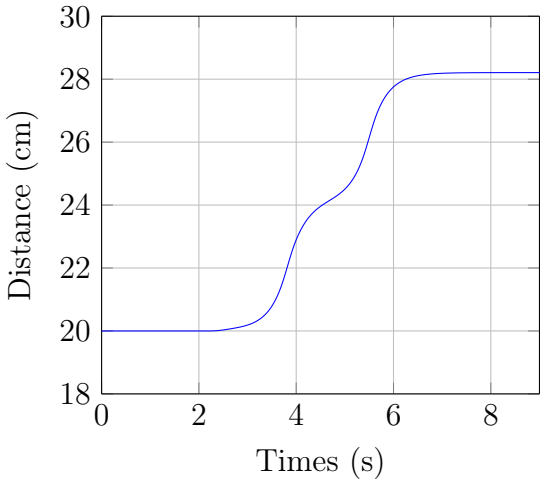
Robustness to Planned Error

This situation represents the case where the robots cannot maintain a constant distance between them because of the environment. For example, the robot may need to get close or farther from each other to access a particularly restricted area. To simulate this case, the robot 2 executes bigger footsteps which results in an increased displacement of 2 cm in comparison to its partner as shown in Fig. 5.12. The relative displacement of robot 2 can be figured out by comparing its ZMP to the green dashed line which corresponds to the original position of the ZMP if it had executed symmetrical footsteps with respect to robot 1. Again, the results show that the robots are able to follow their respective reference

ZMP trajectory despite the interaction force that is generated. However, in order to do so, both robots move their torsos far away from each other for a total displacement of 8 cm as shown in Fig. 5.12(b). As noted in Section 5.7.3, this behavior might generate problems since the robot arm movement is restricted.



(a) Global ZMP and CoM trajectories in the sagittal plane (x-axis). The dashed green line represents the position of the ZMP of robot 2 if it executed symmetrical footsteps.



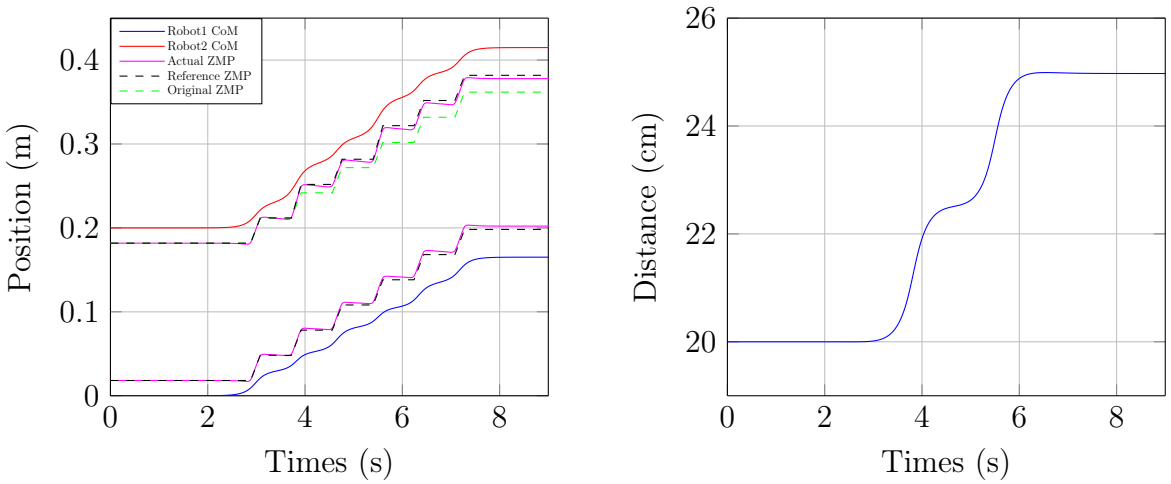
(b) The distance between the robots CoMs during the displacement

Figure 5.12 Simulation where robot 2 executes two footsteps that are 1 cm longer than the footsteps of robot 1.

In order to reduce the torso displacement, we can increase the weight on the relative torso displacement in the minimization objective defined in section 5.7.3. For example, Fig. 5.13 presents the results of the same scenario as Fig. 5.12 except that Q_d was increased from 0 to 10^{-3}. As can be seen, the new objective function effectively reduced the relative torso displacement. An interesting observation is that the relative torso displacement was reduced by about 3 cm, or 1.5 cm per robot while introducing a ZMP tracking error of only 4 mm per robot.

Robustness to disturbance

A challenging problem with controlling the robot-load-robot system is the fact that positional errors are sometimes introduced rapidly in the system. For example, Fig. 5.14 shows the results of an experiment where two Nao robots were placed in front of each other and symmetrical footstep commands were sent to both of them. During the experiment, one of the robots was monitoring its relative distance from the other robot using an RGB-D Camera (Asus Xtion Pro-Live). Fig. 5.14 points out that the robots can't maintain a constant distance from each other even though they received identical



(a) Global ZMP and CoM trajectories in the sagittal plane (x-axis)

(b) The distance between the robots CoMs during the displacement

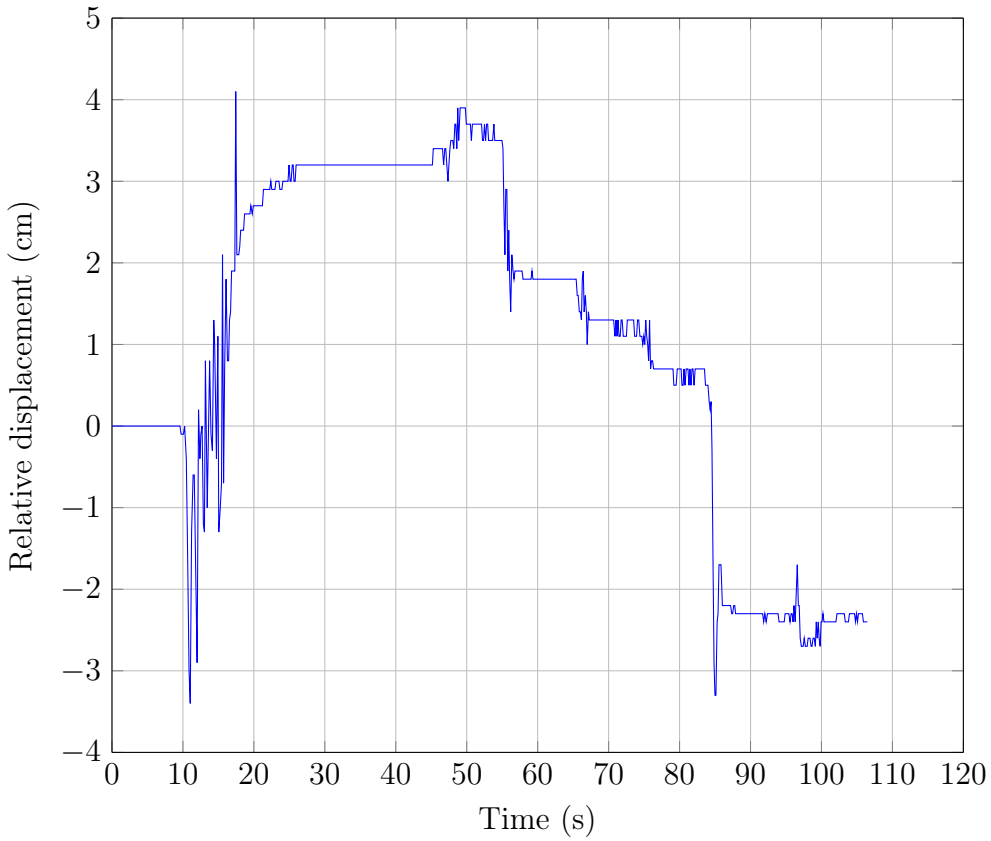
Figure 5.13 Same experiment as in Fig. 5.12 while minimizing the objective function (5.22) instead of (5.18).

commands. However, it is interesting to notice that the error increase incrementally in synchronization with the footsteps, this behavior is expected since the error mostly occurs when the robot foot lands on the floor. This kind of error is really problematic for the controller as instead of handling an error that increases gradually, it should handle relatively big and instantaneous variations of the error.

To simulate this case, let us analyze the controller behavior when an instantaneous error of $2\,cm$ is added to the state of robot 2 (i.e. a $2\,cm$ foot slip). As can be seen in Fig. 5.15, an undesirable motion of the robots CoMs is noticeable as the robots first move their torsos closer to each other, and then rapidly bring them backward. In this simulation, the reference ZMP trajectories are only moved to the new feet positions at the end of the preview window which is about two footsteps. A simple solution was to simply offset the reference ZMP trajectory by the error terms, thereby repositioning it at the current foot position. As can be seen in Fig. 5.16, using this simple solution greatly reduced the undesirable behavior in the CoM motion.

5.10.2 Real Robot Experiments and Results

We conducted experimental validation on a team of two Nao humanoid robots transporting a variety of objects ranging in size and in mass (0-2.13Kg). A selection of representative tests is shown in the video attached to this paper. During all the experi-

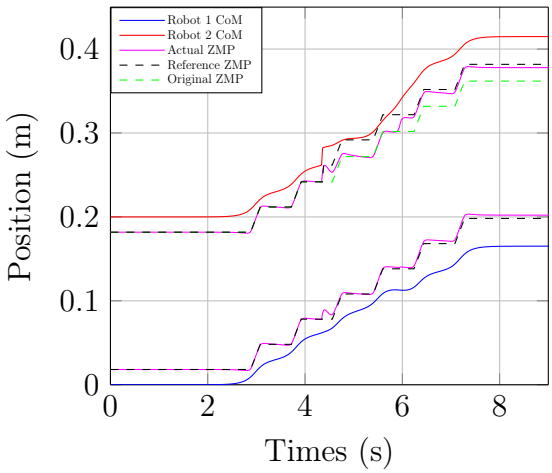


(a) Relative displacement between the robots

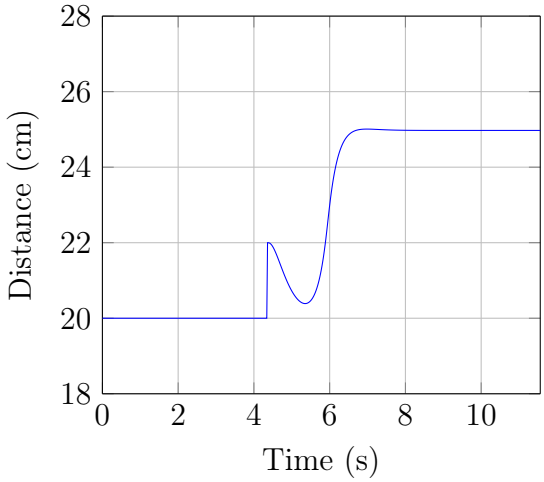
(b) Experimental setup

Figure 5.14 Relative displacement between the two robots while they walk freely and given the same commands.

ments, the robots used a light cardboard table, of about 60 cm in length and 20 cm in width, to carry numerous objects as depicted in Fig. 5.17.

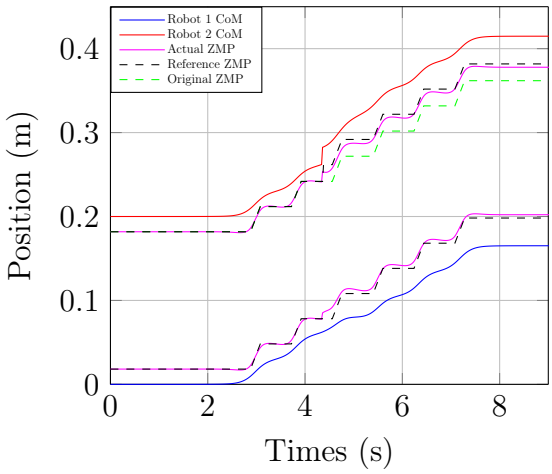


(a) Global ZMP and CoM trajectories in the sagittal plane (x-axis)

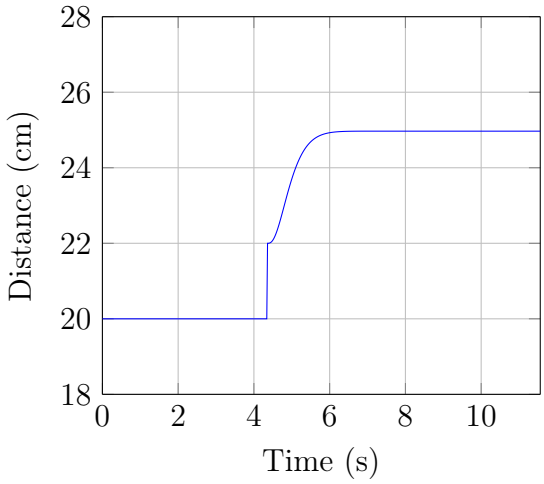


(b) The distance between the robots CoMs during the motion

Figure 5.15 Simulation result for a scenario where an instantaneous positioning error is added to robot 2



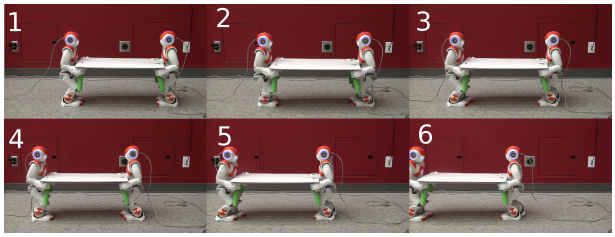
(a) Global ZMP and CoM trajectories in the sagittal plane (x-axis)



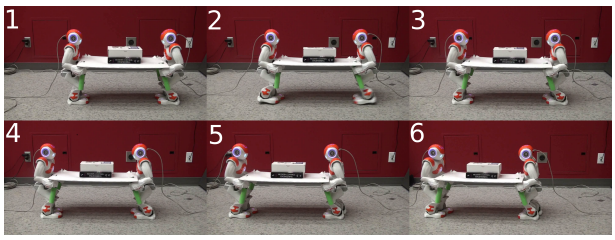
(b) The distance between the robots CoMs during the motion

Figure 5.16 Simulation result for a scenario where an instantaneous position error is added to robot 2 and the ZMP reference trajectory is offset in accordance with the detected error

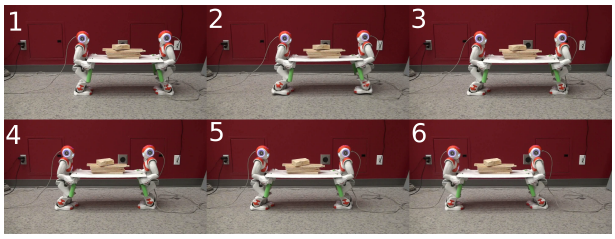
The experiments were carried out at low-speed ($\approx 2cm{\cdot}s^{-1}$) in large part because of the important oscillation in the joints of the robots that occurs when transporting a heavy object. Fig. 5.18 and Fig. 5.19 respectively present the relative displacement between the robots and the robot 1 ZMP during two experiments. As can be seen, the robots displacement is synchronized in both cases as the distance between the robots do not vary



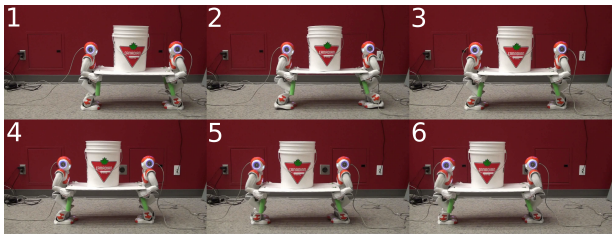
(a) Empty table



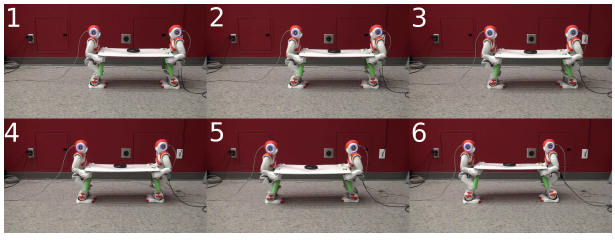
(b) Books



(c) Wood scraps



(d) Large bucket



(e) A mass of 2.23 Kg

Figure 5.17 Experiments with two Nao robots using a table to transport a variety of objects.

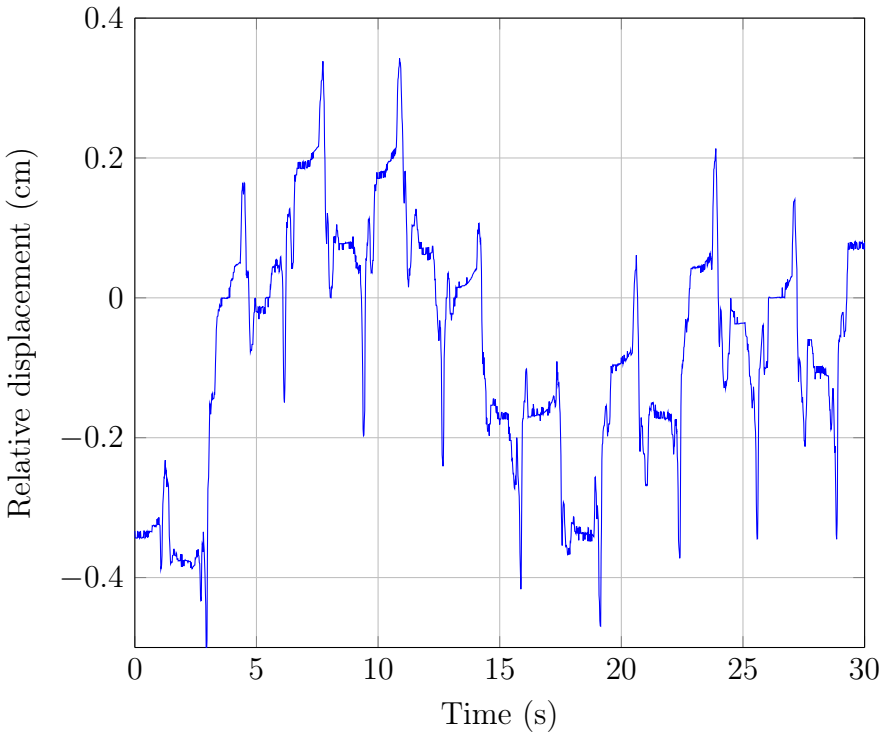
much. Also, the walk is stable as the ZMP is kept well within the support region for most of the motions thereby validating our controller.

A problem that we have encountered during the experiments is that sometimes an important slipping of the robots feet occurred during the execution of the task. Although the stability of the system was always maintained, the speed and performance could certainly be enhanced by developing a stabilizer that is consistent with the controller proposed in this paper. Lastly, as can be seen in the screenshots of the experiments (Fig. 5.17), the table is not completely rigid and it sometimes bent because of the weight and its legs did not stay parallel. As a result, the performance of the arm controller was degraded, and the arms motion range was quite small ($\approx 5cm$).

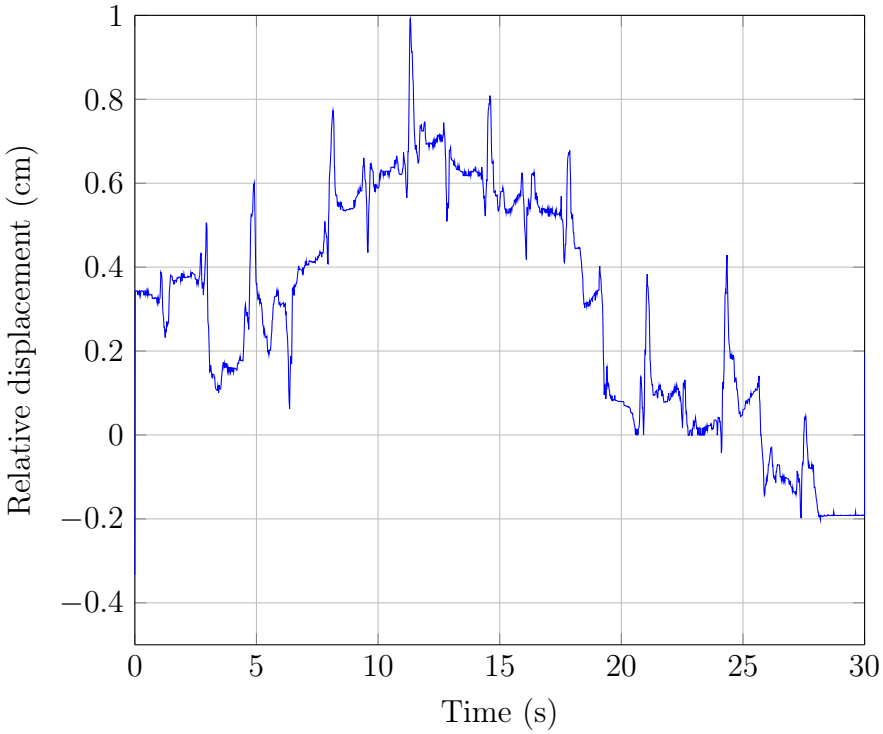
5.11 Conclusion

In this paper, we presented a control framework for the cooperative transportation of a heavy or large object by two humanoid robots. The main building blocks are a local footstep planner, a compliant control of the arms and a ZMP-based pattern generator. The ZMP controller uses a complete dynamic model of the task to ensure the stability of the system. This controller formulation is particularly interesting for smaller humanoid robots in comparison to sensor-based approaches that rely on measurements from force/torque sensors.

Cooperative transportation by humanoids is a topic that has received relatively few attention despite its relevance in multiple scenarios. This article explored both the theoretical and practical side of the problem. Even though the focus was on the actual implementation on the Nao robotic platform, we also discussed how the mass of the transported object and the interaction force model impact the stability of the robots during the execution of the task. These results can be generalized to other humanoid robots and can help in designing a controller that makes the most out of the used platform. Notably, an interesting option would be to unify the arms control and the ZMP preview controller into a single module, thereby providing the ability to modify the interaction model (stiffness coefficient K) to stabilize the system. Another interesting option would be to give the controller the ability to modify the footsteps during execution to increase the responsiveness of the controller. Finally, a limiting factor to the performance of the approach on the Nao was the important joint oscillation and foot slipping due to the physically challenging nature of the task. As mentioned briefly before, we believe the system would greatly benefit from the development of a low-level stabilizer for the Nao and it is a priority for future work.

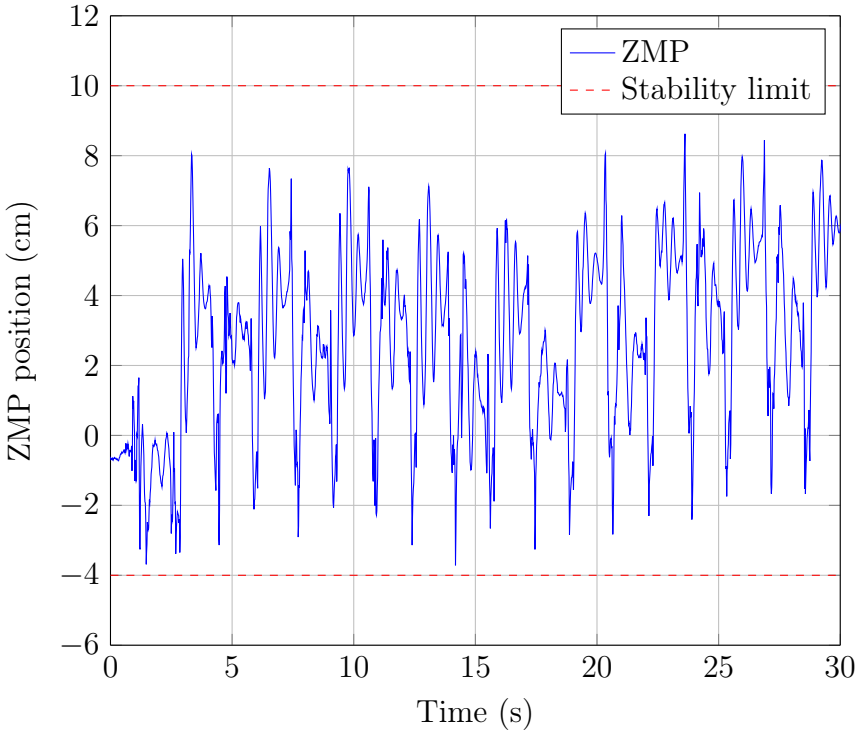


(a) An empty table

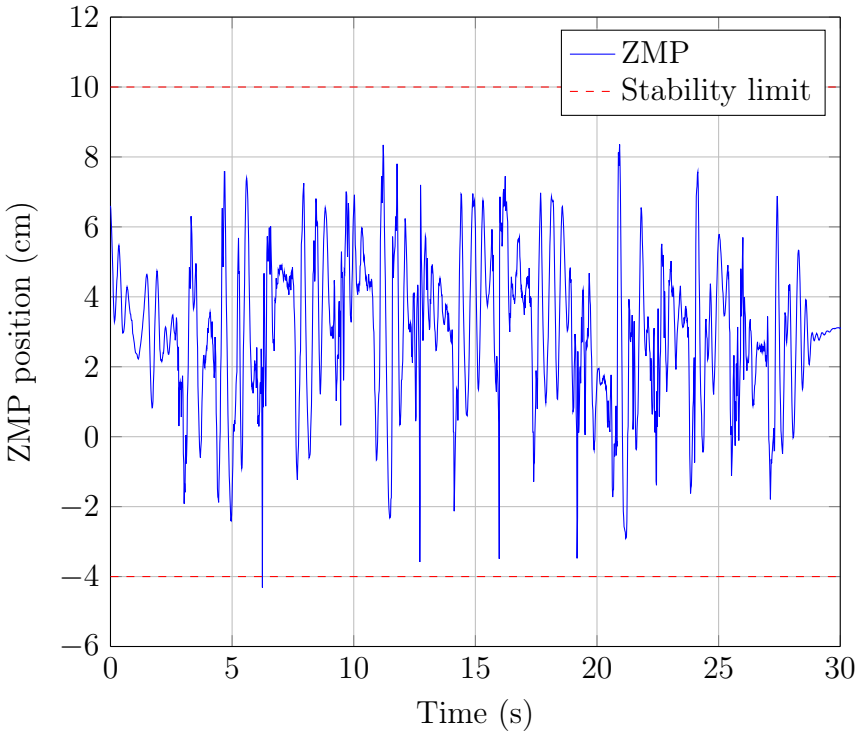


(b) A transported mass of 2.3 Kg

Figure 5.18 The relative displacement of the robots during two experiments



(a) Empty table



(b) A mass of 2.3 Kg

Figure 5.19 Robot 1 ZMP during the two experiments. The stability limits represent the size of the robot foot.

CHAPITRE 6

Conclusion

Dans ce mémoire, des stratégies de contrôle pour le transport d'une charge par un humanoïde utilisant un chariot ainsi que pour le transport collaboratif par deux robots ont été proposées. Lors de tâche de transport, les robots sont soumis à des forces externes causés par l'interaction avec l'environnement et contrairement aux robots à roues, les humanoïdes peuvent être facilement déséquilibrés par ces forces. Ces deux aspects ont été abordés dans le mémoire alors qu'une stratégie d'estimation de force externe a été présenté au chapitre 2 et les chapitres 4 et 5 ont décrit deux contrôleurs permettant la réalisation d'une tâche de (1) transport avec chariot et (2) transport collaboratif par deux robots humanoïdes. Les algorithmes conçus ont été testés sur des robot Nao, des humanoïdes de petite taille , et les résultats confirment la validité de notre approche.

En résumé, les contributions principales de ce projet de maitrise sont :

- Un estimateur de force externe adapté aux humanoïdes de petite taille
- Un contrôleur à faible charge de calculs et simple d'implémentation pour une tâche de transport avec chariot
- Une architecture de contrôle simple d'implémentation permettant à une équipe de deux humanoïdes de transporter une charge importante

Les travaux effectués ont permis de développer un modèle dynamique et un contrôleur permettant à deux humanoïdes de transportant une charge importante de façon stable. Toutefois, certains aspects devraient être intégrés au système pour être plus générale. Notamment, la prise en main de l'objet à transporter, le phase de soulèvement de l'objet ainsi que le dépôt de l'objet à l'endroit désiré sont des sous-tâches qui doivent faire partie d'un système complet de transport collaboratif par robots mais qui n'ont pas été abordés ici. Également, le transport d'objet par des humanoïdes dans une configuration différente de celle abordée dans ce mémoire est un sujet d'intérêt pour le futur.

LISTE DES RÉFÉRENCES

Agravante, D. J., Cherubini, A., Bussy, A., Gergondet, P. et Kheddar, A. (2014). Collaborative human-humanoid carrying using vision and haptic sensing. *IEEE International Conference on Robotics and Automation (ICRA)*, p. 607–612.

Agravante, D. J., Sherikov, A., Wieber, P. B., Cherubini, A. et Kheddar, A. (2016). Walking pattern generators designed for physical collaboration. *IEEE International Conference on Robotics and Automation (ICRA)*, p. 1573–1578.

Aldebaran (2016). Nao official documentation.

Bellaccini, M., Lanari, L., Paolillo, A. et Vendittelli, M. (2014). Manual guidance of humanoid robots without force sensors : Preliminary experiments with NAO. *IEEE International Conference on Robotics and Automation*, p. 1184–1189.

Berger, E., Sastuba, M., Vogt, D., Jung, B. et Ben Amor, H. (2014). Dynamic Mode Decomposition for perturbation estimation in human robot interaction. Dans *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*. p. 593–600.

Berger, E., Vogt, D., Haji-Ghassemi, N., Jung, B. et Amor, H. B. (2015). Inferring guidance information in cooperative human-robot tasks. Dans *IEEE- RAS International Conference on Humanoid Robots*. p. 124–129.

Bouyarmane, K. et Kheddar, A. (2012). Humanoid Robot Locomotion and Manipulation Step Planning. *Advanced Robotics*, volume 26, numéro 10.

Bussy, A., Gergondet, P., Kheddar, A., Keith, F. et Crosnier, A. (2012). Proactive behavior of a humanoid robot in a haptic transportation task with a human partner. Dans *IEEE International Workshop on Robot and Human Interactive Communication*. p. 962–967.

Choi, H., Lee, S., Jin, T. et Lee, S. H. (2015). Trajectory-free reactive stepping of humanoid robots using momentum control. Dans *IEEE-RAS International Conference on Humanoid Robots*. p. 1173–1178.

Don Joven Agravante, Andrea Cherubini, Alexander Sherikov, Pierre-Brice Wieber, A. K. (2017). Human-Humanoid Collaborative Carrying.

Erbatur, K., Okazaki, A., Obiya, K., Takahashi, T. et Kawamura, A. (2002). A study on the zero moment point measurement for biped walking robots. Dans *Advanced Motion Control, 2002. 7th International Workshop on*. p. 431–436.

Ferreau, H., Kirches, C., Potschka, A., Bock, H. et Diehl, M. (2014). qpOASES : A parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, volume 6, numéro 4, p. 327–363.

Garimort, J., Hornung, A. et Bennewitz, M. (2011). Humanoid navigation with dynamic footstep plans. Dans *IEEE International Conference on Robotics and Automation (ICRA)*. p. 3982–3987.

Gouaillier, D., Collette, C. et Kilner, C. (2010). Omni-directional closed-loop walk for NAO. p. 448–454.

Gouaillier, D., Hugel, V., Blazevic, P., Kilner, C., Monceaux, J., Lafourcade, P., Marnier, B., Serre, J. et Maisonnier, B. (2009). Mechatronic design of NAO humanoid. *IEEE International Conference on Robotics and Automation*, p. 769–774.

Griffin, R. J., Wiedebach, G., Bertrand, S., Leonessa, A. et Pratt, J. (2017). Walking Stabilization Using Step Timing and Location Adjustment on the Humanoid Robot, Atlas. Dans *IEEE International Conference on Intelligent Robots and Systems (IROS)*. p. 667–673.

Harada, K., Kajita, S., Kanehiro, F., Fujiwara, K., Kaneko, K., Yokoi, K. et Hirukawa, H. (2007). Real-time planning of humanoid robot's gait for force-controlled manipulation. *IEEE/ASME Transactions on Mechatronics*, volume 12, numéro 1, p. 53–62.

Harada, K., Kajita, S., Kaneko, K. et Hirukawa, H. (2003a). Pushing manipulation by humanoid considering two-kinds of ZMPs. *IEEE International Conference on Robotics and Automation*, p. 1627–1632.

Harada, K., Kajita, S., Kaneko, K. et Hirukawa, H. (2003b). ZMP Analysis for Arm / Leg Coordination. Dans *IEEE/RSJ International Conference on Intelligent Robots and Systems*. p. 75–81.

Harada, K., Kajita, S., Saito, H., Morisawa, M., Kanehiro, F., Fujiwara, K., Kaneko, K. et Hirukawa, H. (2005). A humanoid robot carrying a heavy object. Dans *IEEE International Conference on Robotics and Automation (ICRA)*. p. 1712–1717.

Hawley, L. et Suleiman, W. (2016). External Force Observer for Medium-sized Humanoid Robots. Dans *IEEE-RAS International Conference on Humanoid Robots*. p. 366–371.

Hawley, L. et Suleiman, W. (2017). Control Strategy and Implementation for a Humanoid Robot Pushing a Heavy Load on a Rolling Cart. Dans *IEEE International Conference on Intelligent Robots and Systems (IROS)*. p. 4997–5002.

Hawley, L. et Suleiman, W. (2018). Kalman filter based observer for an external force applied to medium-sized humanoid robots. Dans *Submitted to IEEE International Conference on Intelligent Robots and Systems*.

Herdt, A., Diedam, H., Wieber, P.-B., Dimitrov, D., Mombaur, K. et Diehl, M. (2010). Online Walking Motion Generation with Automatic Footstep Placement. *Advanced Robotics*, volume 24, numéro 5-6.

Hornung, A., Dornbush, A., Likhachev, M. et Bennewitz, M. (2012). Anytime search-based footstep planning with suboptimality bounds. Dans *IEEE-RAS International Conference on Humanoid Robots*. p. 674–679.

Inoue, Y., Tohge, T. et Iba, H. (2003). Cooperative Transportation by Humanoid Robots : Learning to Correct Positioning. *Design and Application of Hybrid Intelligent Systems*, p. 1124–1134.

Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K. et Hirukawa, H. (2003). Biped Walking Pattern Generation by using Preview Control of Zero-Moment Point. Dans *IEEE International Conference on Robotics and Automation (ICRA)*. p. 1620–1626.

Kajita, S. et Tani, K. (1995). Experimental study of biped dynamic walking in the linear inverted pendulum mode. Dans *IEEE International Conference on Robotics and Automation (ICRA)*, volume 3. p. 2885–2891.

Kambayashi, Y., Takimoto, M. et Kodama, Y. (2005). Controlling biped walking robots using genetic algorithms in mobile agent environment. *IEEE 3rd International Conference on Computational Cybernetics, 2005. ICCC 2005.*, p. 29–34.

Kaneko, K., Kanehiro, F., Kajita, S., Hirukawa, H., Kawasaki, T., Hirata, M., Akachi, K. et Isozumi, T. (2004). Humanoid Robot HRP-2. Dans *IEEE International Conference on Robotics and Automation*. p. 1083–1090.

Kaneko, K., Kanehiro, F., Morisawa, M., Yoshida, E. et Laumond, J.-P. (2012). Disturbance observer that estimates external force acting on humanoid robots. *12th IEEE International Workshop on Advanced Motion Control (AMC)*, p. 1–6.

Kim, J. J. et Lee, J. J. (2007). Gait adaptation method of biped robot for various terrains using Central Pattern Generator (CPG) and learning mechanism. *ICCAS 2007 - International Conference on Control, Automation and Systems*, p. 10–14.

Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I. et Osawa, E. (1997). RoboCup. Dans *Proceedings of the first international conference on Autonomous agents*. p. 340–347.

Magrini, E., Flacco, F. et De Luca, A. (2014). Estimation of contact forces using a virtual force sensor. Dans *IEEE International Conference on Intelligent Robots and Systems (IROS)*. p. 2126–2133.

Magrini, E., Flacco, F. et De Luca, A. (2015). Control of generalized contact motion and force in physical human-robot interaction. Dans *IEEE International Conference on Robotics and Automation (ICRA)*. p. 2298–2304.

Mattioli, T. et Vendittelli, M. (2017). Interaction Force Reconstruction for Humanoid Robots. *IEEE Robotics and Automation Letters*, volume 2, numéro 1, p. 282–289.

McGill, S. G. et Lee, D. D. (2011). Cooperative humanoid stretcher manipulation and locomotion. *IEEE-RAS International Conference on Humanoid Robots*, p. 429–433.

Miyata, N., Ota, J., Aiyama, Y., Sasaki, J. et Arai, T. (1997). Cooperative transport system with regrasping car-like mobile robots. Dans *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3. p. 1754–1761.

Motahar, M. S., Veer, S., Huang, J. et Poulakakis, I. (2015). Integrating dynamic walking and arm impedance control for cooperative transportation. Dans *IEEE International Conference on Intelligent Robots and Systems (IROS)*. p. 1004–1010.

Nozawa, S., Ishida, M., Ueda, R., Kakiuchi, Y., Okada, K. et Inaba, M. (2011a). Full-body motion control integrated with force error detection for wheelchair support. *IEEE-RAS International Conference on Humanoid Robots*, p. 193–198.

Nozawa, S., Kakiuchi, Y., Okada, K. et Inaba, M. (2012). Controlling the planar motion of a heavy object by pushing with a humanoid robot using dual-arm force control. *Proceedings - IEEE International Conference on Robotics and Automation*, p. 1428–1435.

Nozawa, S., Maki, T., Kojima, M., Kanzaki, S., Okada, K. et Inaba, M. (2008). Wheelchair support by a humanoid through integrating environment recognition, whole-body control and human-interface behind the user. Dans *IEEE/RSJ International Conference on Intelligent Robots and Systems*. p. 1558–1563.

Nozawa, S., Ueda, R., Kakiuchi, Y., Okada, K. et Inaba, M. (2011b). Sensor-based integration of full-body object manipulation based on strategy selection in a life-sized humanoid robot. *Journal of Robotics and Mechatronics*, volume 23, numéro 2, p. 239–248.

Oriolo, G., Paolillo, A., Rosa, L. et Vendittelli, M. (2015). Vision-based trajectory control for humanoid navigation. *IEEE-RAS International Conference on Humanoid Robots*, p. 118–123.

Ota, J., Miyata, N., Arai, T., Yoshida, E., Kurabatashi, D. et Sasaki, J. (1995). Transferring and regrasping a large object by cooperation of multiple mobile robots. Dans *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3. p. 543–548.

Prahlad, V., Dip, G. et Meng-Hwee, C. (2008). Disturbance rejection by online ZMP compensation. *Robotica*, volume 26, numéro 01, p. 9–17.

Rioux, A. (2016). *Transport coopératif d'un objet par deux robots humanoïdes dans un environnement encombré*. Mémoire, Université de Sherbrooke, Sherbrooke, Canada.

Rioux, A., Esteves, C., Hayet, J.-B. et Suleiman, W. (2015). Cooperative SLAM-Based Object Transportation by Two Humanoid Robots in a Cluttered Environment. Dans *IEEE-RAS International Conference on Humanoid Robots*. p. 331–337.

Rioux, A., Esteves, C., Hayet, J.-B. et Suleiman, W. (2017). Cooperative Vision-Based Object Transportation by Two Humanoid Robots in a Cluttered Environment. *International Journal of Humanoid Robotics*, volume 14, numéro 03, p. 1–30.

Rioux, A. et Suleiman, W. (2015). Humanoid navigation and heavy load transportation in a cluttered environment. Dans *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. p. 2180–2186.

Rioux, A. et Suleiman, W. (2018). Autonomous SLAM based humanoid navigation in a cluttered environment while transporting a heavy load. *Robotics and Autonomous Systems*, volume 99, p. 50–62.

Sakagami, Y., Watanabe, R., Aoyama, C., Matsunaga, S., Higaki, N. et Fujimura, K. (2002). The intelligent ASIMO : system overview and integration. Dans *IEEE/RSJ International Conference on Intelligent Robots and System (IROS)*, volume 3. p. 2478–2483.

Sato, T. et Ohnishi, K. (2008). ZMP disturbance observer for walking stabilization of biped robot. Dans *International Workshop on Advanced Motion Control, AMC*, volume 1. p. 290–295.

Scharf, D., Hadaegh, F. et Ploen, S. (2004). A survey of spacecraft formation flying guidance and control. Part II : control. Dans *American Control Conference*, volume 4. p. 2976–2985.

Shim, B. G., Lee, E. H., Min, H. K. et Hong, S. H. (2004). Study of Zmp Measurement for Bipedrobot Using Fsr Sensor. Dans *21st International Symposium on Automation and Robotics in Construction*.

Stasse, O., Evrard, P., Perrin, N., Mansard, N. et Kheddar, A. (2009). Fast foot prints re-planning and motion generation during walking in physical human-humanoid interaction. Dans *IEEE-RAS International Conference on Humanoid Robots*. p. 284–289.

Stephens, B. (2008). Humanoid push recovery. Dans *IEEE-RAS International Conference on Humanoid Robots*. p. 589–595.

Vukobratović, M. et Borovac, B. (2004). Zero-Moment Point — Thirty Five Years of Its Life. *International Journal of Humanoid Robotics*, volume 01, numéro 01, p. 157–173.

Wieber, P. B. (2006). Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. Dans *IEEE-RAS International Conference on Humanoid Robots*. p. 137–142.

Wu, M. H., Konno, A., Ogawa, S. et Komizunai, S. (2014). Symmetry cooperative object transportation by multiple humanoid robots. *IEEE International Conference on Robotics and Automation (ICRA)*, p. 3446–3451.

Wu, M.-H., Ogawa, S. et Konno, A. (2016). Symmetry position/force hybrid control for cooperative object transportation using multiple humanoid robots. *Advanced Robotics*, volume 30, numéro 2, p. 131–149.

Xinjilefu, X., Feng, S. et Atkeson, C. G. (2015). Center of Mass Estimator for Humanoids and its Application in Modelling Error Compensation, Fall Detection and Prevention. *IEEE-RAS International Conference on Humanoid Robots*, p. 67–73.

Yokoyama, K., Handa, H., Isozumi, T., Fukase, Y., Kaneko, K., Kanehiro, F., Kawai, Y., Tomita, F. et Hirukawa, H. (2003). Cooperative works by a human and a humanoid robot. Dans *IEEE International Conference on Robotics and Automation (ICRA)*. p. 2985–2991.